

MONITOR GRAFICO  
CARACTERES NO COLOR

1986 C&M MAXX

# Micro Sistemas

A PRIMEIRA REVISTA BRASILEIRA DE MICROCOMPUTADORES



JOGOS: Programação  
Editor de Adventures

1255



# GRAFIX 80 F/T. COMPATÍVEL ATÉ COM SEU PAPEL DE CARTA.



A GRAFIX 80 F/T é versátil. Com ela você pode facilmente imprimir formulários contínuos e folhas soltas como papéis de carta, envelopes, etiquetas, etc. Isto porque a GRAFIX 80 F/T é a única que vem equipada com trator e com cilindro de borracha, igual aos das máquinas de escrever, permitindo a impressão da primeira à última linha.



A GRAFIX 80 F/T é rápida e

silenciosa. Imprime em 80 colunas a 160 caracteres por segundo com baixo nível de ruído.

Além de tudo isso, a GRAFIX 80 F/T é compatível com todos os microcomputadores, especialmente com os compatíveis IBM-PC, imprimindo todos os softwares disponíveis (Dbase III, Lotus 1-2-3, Word, ABC, Chartstar etc.)

Conheça a GRAFIX 80 F/T. A impressora certa para seu micro pessoal ou profissional.

## SCRITTA

Al. Amazonas, 832 - Alphaville  
Barueri - SP Tel. 421-1247

**GRAFIX. TECNOLOGIA SEMPRE EM EVOLUÇÃO.**



**N**ossa edição de JOGOS traz diversos programas e idéias interessantes, mesmo para aqueles usuários que preferem aprender a fazer do que jogar. Pensando neles, inserimos nesta edição, além das listagens de jogos, artigos que contêm dicas preciosas para os curiosos e autodidatas.

No artigo "A programação de jogos", nosso colaborador Giangiacomo Ponzo Neto percorre, com exemplo, os diversos elementos que devem constar de um bom jogo, dando sugestões de como melhor explorar os recursos das máquinas. Já Paolo Fabrizio Pugno colaborou neste número com um editor de adventures, que permitirá a você criar suas próprias aventuras. Nesta classe de jogos, um artigo deste tipo é especialmente aconselhável, visto serem os adventures programas em que o criador pode gerar enredos, cenários e personagens com vida própria. Pode ser uma excelente experiência você criar um jogo para seu amigo tentar! E além das inúmeras outras atrações que este número de MS lhe traz, os usuários da linha ZX Spectrum poderão aprender — e muito — sobre a manipulação de seu vídeo em Assembler e a geração de gráficos, se tiverem o trabalho de digitar o inédito monitor gráfico para o TK90X, de autoria do programador Divino C. R. Leitão, criador de diversos jogos comerciais, como o Walkirie, Q\*BERT, Cavernas de marte e Pinball, pela marca Ciberne, e que já há alguns números integra a equipe de MS, na coordenação do CPD.

### NOTA AOS LEITORES DE MS

• A partir desta edição, MICRO SISTEMAS adotará uma nova classificação no material publicado para a linha Sinclair. Na categoria ZX81, estarão os programas e dicas compatíveis com os micros nacionais TK82, TK85 e CP-200. Quando falarmos em ZX Spectrum, estaremos nos referindo aos programas que rodam no TS2068 e no TK90X (por hora, nosso único compatível com o Spectrum inglês). Finalmente, por linha Sinclair entenderemos o material que serve tanto para o ZX81 quando para o ZX Spectrum.

*Alda Campos*

P & D Sistemas Eletrônicos S.A.

Biblioteca

# Micro Sistemas

ANO V - N.º 52 - JANEIRO 1986

## SUMÁRIO

- 6** A PROGRAMAÇÃO DE JOGOS — Artigo de Giangiacomo Ponzo Neto, para a linha Sinclair.
- 12** ROLETA — Programa de Antônio José Ávila da Silva, para a linha TRS-80.
- 14** MONITOR GRÁFICO — Artigo de Divino C. R. Leitão, para a linha ZX Spectrum.
- 20** CARACTERES NO COLOR — Artigo de Cláudio Costa.
- 24** EDITOR DE ADVENTURES — Programa de Paolo Fabrizio Pugno, para a linha ZX Spectrum.
- 50** ALTA RESOLUÇÃO EM ASSEMBLER (I) — Artigo de Evandro Mascarenhas de Oliveira, para a linha Apple.
- 56** CURSO DE FORTH (IV) — Série de autoria do professor Antonio Costa.

### BANCO DE SOFTWARE

- 30 THUNDERBIRD
- 32 FUGA CÔSMICA
- 36 ÓRBITA
- 38 BATALHA NAVAL
- 40 OLHADINHA
- 42 POLARIS
- 44 O PEIXE GULOSO
- 46 RALLY
- 48 MICRO-GOLF

### SEÇÕES

4 CARTAS

53 LIVROS

18 BITS

54 DICAS

CAPA: José Herádio Silva



# Micro Sistemas

EDITORIA / DIRETORA RESPONSÁVEL:  
Aida Surerus Campos

ASSESSORIA TÉCNICA:  
Roberto Ouito de Sant'Anna; José Eduardo Neves; Luiz Antonio Pereira; Claudio José Costa.

CPD: Divino C. R. Leitão, Pedro Paulo Pinto Santos.

REDAÇÃO: Graça Santos (Subeditoria); Stella Lachter-macher; Mônica Alonso Monçores; Carlos Alberto Azevedo.

COLABORADORES: Aldo Neletto Jr.; Alvaro de Filippo; Amaury Moraes Jr.; André Koch Zielasko; Antonio Costa Pereira; Ari Morato; Cláudio de Freitas Bittencourt; Evandro Mascarenhas de Oliveira; Geraldo Simonetti Bello; Heber Jorge da Silva; Ivan Camilo da Cruz; João Antônio Zuffo; João Henrique Volpini Mettos; José Carlos Niza; José Ribeiro Pene Neto; Lívio Pareschi; Luciano Nilo de Andrade; Luis Carlos Eiras; Luiz Carlos Nardy; Marcel Gemeleira de Albuquerque; Marcel Tarrisse da Fontoura; Maurício Costa Reis; Marcelo Renato Rodrigues; Mário José Bittencourt; Nelson Hisashi Temura; Nelson N. S. Santos; Newton D. Braga Jr.; Paulo Sérgio Gonçalves; Rizeri Maglio; Rudolph Horner Jr.; Sérgio Veludo.

ARTE: Cláudia Duarte (coordenação); Leonardo Santos (diagramação); Maria Christina Coelho Marques (revisão); Wellington Silveira e Marco Antônio Rubim (arte-final).

ACOMPANHAMENTO GRÁFICO: Fábio da Silva

ADMINISTRAÇÃO: Janete Sarno

## PUBLICIDADE

São Paulo:  
Geni dos Santos Roberto  
Contatos: Paulo Gomide;  
Tels: (011) 853-3229, 853-3152

Rio de Janeiro:  
Elizabeth Lopes dos Santos  
Contatos: Regina Gimenez; Georgina de Oliveira  
Tel: (021) 262-6306

Minas Gerais:  
Sidney Domingos da Silva  
Rua de Bahia, 1148 — sala 1318  
CEP 30.000 — Belo Horizonte

Porto Alegre:  
AURORA-ASSESSORIA EMPRESARIAL LTDA.  
Rua dos Andradas, 1155 - Grupo 1606 - 16º andar  
Tel.: (0512) 26-0839

CIRCULAÇÃO E ASSINATURAS:  
Ademer Belon Zochio (RJ)  
José Antônio Alarcon (SP) - Tel.: (011) 853-3800

COMPOSIÇÃO:  
Studio Alfa, Coopim.

FOTOLITO:  
Ingrafoto, Reprolito e Juracy Freire

IMPRESSÃO:  
JB Indústrias Gráficas

DISTRIBUIÇÃO:  
Fernando Chinaglia Distribuidora Ltda.  
Tel: (021) 268-9112

ASSINATURAS:  
No país: 1 ano: Cr\$ 140.000

NÚMEROS ATRASADOS  
Cada exemplar: Cr\$ 8.000

Os artigos assinados são de responsabilidade única e exclusiva dos autores. Todos os direitos de reprodução do conteúdo da revista estão reservados e qualquer reprodução, com finalidade comercial ou não, só poderá ser feita mediante autorização prévia. Transcrições parciais de trechos para comentários ou referências podem ser feitas, desde que sejam mencionados os dados bibliográficos de MICRO SISTEMAS. A revista não aceita material publicitário que possa ser confundido com matéria redacional.



MICRO SISTEMAS é uma publicação mensal da



Análise, Teleprocessamento e Informática Editora Ltda.

## Endereços:

Rua Oliveira Dias, 153 - Jardim Paulista - São Paulo/SP - CEP 01433 - Tels: (011) 853-3800 e 881-5668 (redação)

Av. Presidente Wilson, 165 - grupo 1210 - Centro - Rio de Janeiro/RJ - Tel: (021) 262-6306

## cartas

O sorteado deste mês, que receberá uma assinatura anual da revista MICRO SISTEMAS, é Ernst R. A. Bergmann, do Rio de Janeiro - RJ.

## MAPA DA ROM

Tenho algumas dúvidas com relação ao interessante artigo *Apple: O Mapa da ROM*, encerrado em MS nº 46. São elas:  
a) Consta que o interpretador está na área \$D000-\$D7FF; está correto?  
b) Nas linhas \$45-\$49 estão os registradores A, X, Y, P e S ou A, X, Y, S e P, respectivamente?  
c) Na área usada pelo BASIC, \$0800-\$BFFF, não constou que as áreas \$0C00-\$1FFF e \$6000-\$BFFF não são usadas e que estão livres para o usuário.  
Eduardo Bonaz Marques  
São Paulo — SP

De acordo com nosso procedimento habitual, enviamos sua carta ao autor do artigo e recebemos a seguinte resposta:

"Verifiquei com cuidado os pontos que você ressaltou em sua carta. Com relação ao primeiro item, realmente houve falha de minha parte: o interpretador BASIC APPLE-SOFT fica entre \$D000 e \$F7FF, e não \$D7FF. A respeito dos outros dois pontos, porém, não há erro nenhum. Tudo indica que suas dúvidas com relação à ordem de armazenamento dos registradores e às áreas livres na memória são originadas do Guia do Usuário do Apple. Lá, os autores se embanaram com os registradores e chegaram a conclusões erradas, invertendo o stack pointer e o registrador de flags (S e P). A ordem correta é: A, X, Y, P e S.

Sobre as áreas \$C00-\$1FFF e \$6000-\$BFFF, cuidado! Elas não estão livres para o usuário! O programa BASIC e suas variáveis começam em \$800 e muito frequentemente vão além de \$C00, chegando mesmo a atropelar as páginas gráficas de alta resolução. Do outro lado, as strings começam a ser armazenadas logo abaixo do HIMEM: e vêm descendo em direção ao início da memória. Neste processo de fechamento da memória, o BASIC não respeita nada, nem mesmo suas páginas gráficas — você deve protegê-las através dos comandos HIMEM: e/ou LOMEM: . Já que entramos no assunto, cuidado com a explicação que é dada no livro para o LOMEM: (LOMEM: serve para definir o endereço em que vão começar a ser armazenadas as variáveis — o interpretador executa automaticamente um LOMEM: para a primeira posição livre após o programa)."  
Aldo Felício Naletto Júnior  
São Paulo — SP

## LIVROS/BAS

No programa Livros/Bas, publicado em MS nº 36, a partir do registro 100 do Arquivo Livros, a listagem feita na tela aparecia com uma linha em branco entre cada registro listado. Para resolver este problema, fiz algumas pequenas modificações nas linhas 610 e 1560. As modificações se resumem em acrescentar, no início das duas linhas K\$=STR\$(K); e, no final de ambas, trocar K por K\$.  
Sérgio Veludo  
Belém — PA

## SOS AOS LEITORES

Ao estudar Assembler no CP 400, deparei-me com o problema de geração de som e verifiquei que para produzi-lo é necessário preparar o Sel 1 e o Sel 2 e variar a porta de som (\$FF20), gerando assim sons. Meu problema reside em como controlar os seguintes parâmetros dos sons produzidos: frequência, duração e volume. Agradeço qualquer ajuda. As cartas podem ser enviadas para a Rua Arlindo Nogueira, 1165/sul, CEP 64000. Leonardo Robert de C. Braga  
Teresina — PI

## MS AGRADECE

Esta é a primeira vez que escrevo a MS, apesar de ser assinante há três anos. Desejo congratulá-los pelo curso de FORTH, iniciado em MS nº 49, o qual gostei muito. Desejaria agora que fosse publicado um artigo sobre os diversos compiladores FORTH. Gostaria também que fosse publicada alguma dica ou programa para destravar os programas do TK90X.  
Sotero Luiz da Silva  
Santos — SP

## CORRESPONDÊNCIAS

Desejo entrar em contato com os Cocomanufacos de São Paulo. Correspondências para a Rua Manuel Alonso Medina nº 207, CEP 04160, Jardim Umuarama.  
J. Ronaldo Bressane Júnior (São Paulo — SP)

Gostaria de trocar informações, dicas e programas com usuários do ZX Spectrum. Meu endereço é SHIN QI — 11, cj. 13, casa 07, CEP 71500.  
Alexandre M. Brito (Brasília — DF)

## PONTO DE VISTA

Li com muito interesse a reportagem *Software: um produto nacional bruto*, em MS nº 48. No box sobre produção amadora, V. Sas. dizem que as idades dos programadores variam de 16 até 50 anos. Isto não pode ser uma verdade.

Olhem para mim: sou programador amador, muito ativo e sempre estudando. Iniciei minha carreira de programador no ano de 1982, com um NE-Z8000. Aprendi BASIC como autodidata, exatamente da maneira descrita na matéria. Além disso, fabriquei, entre outras coisas, um teclado, bem como o monitor de vídeo publicado em MS nº 21. No trabalho, uso os computadores ao máximo.

Hoje tenho, além do NE-Z, um TK 2000, um RINGO, alguns filhinhos pré-programadores, uma família encantada com os programas, uma mulher com ciúmes do micro, uma memória melhor do que nunca e 64 anos bem contados.

Eu acho que, na verdade, as idades dos programadores variam de 16 (?) até 80 anos.

Ernst R. A. Bergmann  
Rio de Janeiro — RJ

Envie sua correspondência para:  
ATI - Análise, Teleprocessamento e Informática Editora Ltda., Av. Presidente Wilson, 165/gr. 1210, Centro, Rio de Janeiro/RJ, CEP 20030, Seção Cartas/Redação MICRO SISTEMAS.

"BOM, É QUE NÃO CAI NA MONOTONIA. AQUI MESMO NA EMPRESA EU FAÇO CURSOS DE APRIMORAMENTO"

Fernando Lopes  
Técnico de Manutenção



"O PLIM PLIM DA TV PARECIA O BIP. AÍ EU SAÍA CORRENDO"

Rogério de Figueiredo  
Técnico Pleno



"NUNCA DEIXAMOS UMA MÁQUINA PARADA. SEJA QUAL FOR, ONDE FOR, VAMOS CONSERTAR!"

Mauro Diniz  
Auxiliar Técnico

"MINHA MULHER NÃO ACORDA MAIS COM O BIP. EU VOU E VOLTO, E ELA NEM PERCEBE"

Luís Alberto Silva  
Técnico de Manutenção



## CONCERTO E MANUTENÇÃO DE COMPUTADORES

021 240-4934  
011 240-5899

**MICROSERVICE**  
SUPORTE DE INFORMÁTICA

# COLOQUE O SEU PROGRAMA

SH Publicidade

RETURN

## O MTS-IV-M RESOLVE.

O MTS-IV-M é um Micro Multiusuário, admite até 6 terminais de vídeo teclado com capacidade de executar múltiplas tarefas. Digite.

**MAQUIS** COMPUTADORES

Rio de Janeiro — Av. Calógeras, 6-B — Sobrelojas  
Tels.: 240-4934 220-9943 — Castelo TWX (021) 30354  
CEP: 20030

São Paulo — Rua Inhambu, 1.271 — Tel.: 240-5899  
Moema TWX (011) 21299  
CEP 04520

Dê uma chance a sua criatividade. Com um micro da linha Sinclair, algumas técnicas e bom-senso, você poderá fazer seus jogos!

# A PROGRAMAÇÃO DE JOGOS

Giorgio Ponzio Neto

Neste artigo, trataremos dos elementos constituintes de um jogo e sua elaboração, a partir de conceitos básicos, procurando exemplificá-los para um melhor entendimento. Para isso, embora o artigo procure ser o mais eclético possível, usaremos o BASIC Sinclair, por ser uma linguagem de fácil conversão e uso bastante difundido.

Dividiremos o assunto em dois itens principais: **Rotinas Básicas**, comuns a quase todos os jogos animados e que dão a "base" do programa; e **Recursos Adicionais e Sofisticações**, elementos que, mesmo não sendo obrigatórios, podem enriquecer um jogo, tornando-o mais atrativo.

A primeira coisa a se pensar é, evidentemente, a idéia do jogo. Após ter algo planejado, é uma boa prática desenvolver as situações imaginadas na tela, para se ter noção de como será o visual do que imaginamos. Assim, é necessário desenhar as personagens envolvidas numa disposição semelhante a que se deverá ter, mesmo que ainda não exista movimento, o que ajudará inclusive a aprimorar o pensamento inicial. A partir daí, passaremos às Rotinas Básicas, com a descrição das mais comuns.

## DESENHO DOS COMPONENTES

Existem dois tipos de desenho num jogo animado: os fixos, que formam o cenário; e os móveis, que podem sofrer deslocamentos ou sumir e/ou aparecer. Os primeiros são os mais simples para se trabalhar (o que não implica em menor complexidade), e deve-se procurar fazê-los usando o tipo de comando mais fácil da linguagem, já que devem ser definidos no início do jogo e o loop básico do programa não deve passar por eles; não importando, assim, a eventual lentidão dos comandos usados. Nestes desenhos incluem-se paisagens de fundo, solos e outras figuras que permanecerão inalteradas.

O segundo tipo, desenhos móveis, são os que mais requerem cuidados. Sua impressão, em constante mudança de posição, deve ser rápida o suficiente para não comprometer a emoção do jogo.

Temos dois modos usuais de desenhar na tela: através de comandos do tipo **PRINT**, **PLOT**, etc., ou com uso direto do arquivo de imagem, usando **POKE**. O primeiro é mais simples, e apresenta vantagem, em termos de velocidade, quando o que temos a imprimir, de uma só vez, não é muito pequeno (mais de quatro caracteres). Isto porque, apesar do **POKE** ser mais rápido, ele só imprime um caráter por vez, sendo assim necessários vários deles para formar uma figura maior.

Desenhar com **PRINT** ou **PLOT** é elementar e só vai depender de uma boa imaginação; porém, a impressão com **POKE** é um pouco mais complexa. A grande maioria dos micros têm o

vídeo fixamente mapeado na memória. Neste caso, o **POKE** leva o endereço do ponto de tela que se quer acessar. No ZX81, entretanto, a memória de vídeo tem início variável, dependendo do tamanho do programa. Para encontrar seu início, devemos proceder como no exemplo abaixo:

```
10 LET A=PEEK 16396+256*PEEK 16397+1
20 POKE A,129
30 POKE A+1,138
40 POKE A+2,131
```

O programa anterior imprime uma "nave" nas três primeiras posições da tela. A linha 10 encontra o endereço da primeira posição. Cada linha de tela tem 32 caracteres e mais um **NEW-LINE** (código 118) na posição 33. Para imprimir a nave na segunda linha da tela, poderíamos fazer:

```
50 POKE A+33,129
60 POKE A+34,138
70 POKE A+35,131
```

É interessante lembrar que, uma vez encontrado o valor na linha 10, esta não precisa mais ser lida, desde que não se perca a variável (no caso: A). Portanto, a linha citada só precisa figurar na inicialização do programa.

Para maiores detalhes e dicas sobre desenhos na tela, veja o artigo "Arte e Técnica na Tela do Computador" (MS nº 27).

## MOVIMENTAÇÃO DE PERSONAGENS

As figuras móveis são os elementos que representam a ação do jogo. Para tal, elas devem ser impressas de forma rápida e movidas com igual preocupação.

Para ilustrar, veremos a maneira mais comum de movimento, a partir de um caso geral. Para mover uma figura em qualquer direção, da posição X para Y, teremos, basicamente, o mesmo método: dada a figura em X, primeiro a apagamos e depois a imprimimos em Y (estes dois passos podem ser trocados). Exemplificando:

```
10 LET A=0
20 LET B=0
30 PRINT AT A,B;"S"
40 LET A=A+1
50 LET B=B+1
60 PRINT AT A-1,B-1;" "
70 GOTO 10
```

Para termos movimento em outra direção e/ou sentido basta alterar os números nas linhas 40, 50 e 60. É válido notar que alguns micros não usam a formatação de tela por linha e coluna. É o caso dos TRS-80. Para o esclarecimento desta e de outras diferenças, é muito interessante ler o artigo "Dialeto BASIC" (MS nº 46).

Não se pode, em BASIC, melhorar muito a velocidade des-

se movimento. Existem, porém, algumas exceções. Por exemplo, para um movimento horizontal na linha 21 da tela, poderíamos alterar o procedimento anterior com as seguintes modificações, deletando as linhas 10 e 40 e mantendo as outras:

```
30 PRINT AT 21,B;"S"
60 PRINT AT 21,B-1;" "
```

Entretanto, para este caso podemos usar outra rotina, bem mais simples e rápida:

```
10 LET B=0
20 PRINT AT 21,B;"S"
30 LET B=B+1
40 GOTO 20
```

Agora não há necessidade de se apagar a figura anterior, pois o espaço em branco deixado na string da linha 20 já o faz. Existe ainda a possibilidade de se usar o **POKE**, para uma eventual aceleração da rotina.

Uma técnica de simulação de maior velocidade é a de se imprimir a figura em mais de uma posição, além de seu local de origem. Equivaleria em substituir a rotina anterior por:

```
10 LET B=0
20 PRINT AT 21,B;"S"
30 LET B=B+2
40 GOTO 20
```

Este artifício pode ser usado para obtermos diferentes velocidades para cada figura. Perceba que o número somado na linha 30 é sempre igual ao número de espaços em branco antes da figura. Neste caso só tem sentido haver espaços do lado esquerdo, já que o movimento é só para a direita. Caso a figura também se movesse para a esquerda ou em ambos os sentidos seriam necessários ainda espaços no lado direito.

Como idéia final, ilustraremos também um caso de movimento oscilatório:

```
10 PRINT AT 10,10;"■■■";AT 10,10;"SOS"
20 FOR F=1 TO 4
30 NEXT F
40 GOTO 10
```

Esse efeito dá a sensação de superposição de imagens, podendo ser utilizado para realçar textos ou figuras ou na simulação de alguns eventos, como a movimentação das hélices de um helicóptero, por exemplo. O loop entre as linhas 20 e 30 serve apenas para definir uma espera.

## ORDENS DO JOGADOR

Tudo que depende do jogador no desenrolar do jogo decorre do pressionamento de teclas específicas ou através de um joystick. Daí a necessidade de comandos que interfiram no processamento sem haver parada e alteração do programa. Na verdade, o ideal é que, além dele não ter que ser alterado, ele não pare. Isso limita muito o uso do **INPUT**, por ele interromper o andamento do jogo. A maneira usual de interagirmos com o programa é usar o comando **INKEY\$** (GET em alguns micros). Um primeiro modo de se tentar usar o **INKEY\$** para duas ou mais ações, decorrendo da mesma tecla, poderia ser realizado pela seguinte rotina:

```
10 IF INKEY$="0" THEN PRINT "A";
20 IF INKEY$="0" THEN PRINT "B";
30 GOTO 10
```

Duas ações distintas (no caso, escrever A e B na tela) decorrem do pressionamento da tecla 0. A rotina anterior funciona, porém havendo um inconveniente: como não há garantia de que você vai pressionar o 0 exatamente quando o processamento for para a linha 10, tirando o dedo da tecla na hora em que ele sair da linha 20, é impossível ter a certeza de que as ações serão cumpridas na ordem estabelecida. Isto é resolvido no exemplo abaixo:

```
10 LET AS=INKEY$
20 IF AS="0" THEN PRINT "A";
30 IF AS="0" THEN PRINT "B";
40 GOTO 10
```

Nesse caso, a hierarquia da sequência será sempre respeitada. Outro método usual de se interagir com o jogo é criar uma "bandeira" (uma variável que sinaliza alguma condição), conforme é mostrado abaixo (a bandeira é X, que fica "ligada" quando pressionamos a tecla 0).

```
10 LET X=0
20 IF INKEY$="0" THEN LET X=1
30 IF X=1 THEN PRINT "A";
40 IF X=1 THEN PRINT "B";
50 GOTO 10
```

Em alguns casos, pode-se querer uma ação no jogo em decorrência de um toque em uma tecla, sem que se precise mantê-la pressionada. Isto se consegue substituindo a linha 50 por **50 GOTO 20** e acrescentando a linha **45 IF INKEY\$="1" THEN LET X=0**. A bandeira agora é usada para sinalizar a condição "desligada", quando pressionamos a tecla 1.

O uso do **INKEY\$** é comum também nos **GOTOs** condicionais, como é exemplificado abaixo:

```
IF INKEY$="0" THEN GOTO XXXX
```

Neste caso, ainda, podemos usar uma bandeira:

```
IF X=0 THEN GOTO XXXX
```

A bandeira já teria sido definida por um comando **INKEY\$**. Um detalhe: os exemplos mostrados são simples, e poderia se perguntar se valeriam a pena os métodos mostrados após o primeiro. Porém, há que se entender que poderíamos ter várias ações complexas, decorrendo do acionamento de teclas, e não só o simples fato de se escrever A e B na tela. Nestes casos, os métodos vistos devem ser considerados criteriosamente.

## TESTES DE COLISÃO

Quando um tiro atinge um avião, ou qualquer outro caso em que o cruzamento ou colisão de figuras ocorre, imediatamente o programa deve proceder a uma modificação na situação. Esta se traduz normalmente em desvios para outras partes do programa, em mudança de condições na tela, incremento ou decremento de contadores etc. Porém, o que interessa, a priori, é detectar as eventuais colisões de figuras através dos testes.

Temos dois modos básicos de fazê-los: diretamente das informações de tela ou através do armazenamento das posições das figuras em variáveis, matrizes ou mesmo em posição mais alta da memória, criando um buffer. No primeiro caso, comparam-se os caracteres das personagens envolvidas com os caracteres diversos e procede-se à ação necessária. No segundo, comparam-se as posições na tela dos elementos que colidiram com o que havia no buffer, e toma-se a decisão programada.

Usaremos um pequeno programa para exemplificar os dois casos. Um míssil surge na tela e tenta acertar uma barreira. Pelo primeiro método teremos que, a cada "passo" do míssil, testar se a posição para a qual ele progride contém algum caráter da barreira. O caráter de fim de tela também pode ser testado para dar fim ao curso do míssil. Uma alternativa a isso é desenvolver um contador dos passos do míssil, que é checado periodicamente para sabermos o fim do tiro. Vamos ao exemplo:

```
10 FOR F=0 TO 21
20 PRINT AT F,31;""
30 NEXT F
40 LET B=0
50 LET A=INT (RND*6)+8
60 PRINT AT 10,30;"X";AT 11,30;"X";AT 12,30;"X"
70 PRINT AT A,B;">";AT A,B+3;
80 LET B=B+1
90 IF PEEK (PEEK 16398+256*PEEK 16399)+61 THEN GOTO 120
100 IF PEEK (PEEK 16398+256*PEEK 16399)+8 THEN GOTO 110
110 GOTO 70
120 PRINT AT 10,12;"ACERTOU"
130 PRINT AT 10,30;"";AT 11,30;"";AT 12,30;""
140 GOTO 160
150 PRINT AT 10,12;"ERROU"
160 PAUSE 80
170 PRINT AT 10,12;"",AT A,B;" "
180 GOTO 40
```

Uma rápida análise: as linhas 40 e 50 definem parâmetros do míssil, as linhas de 10 a 30 e 60 desenham o fim de tela e a barreira (o caráter da linha 20 é **GRAPHICS + SHIFT + A** e o da linha 130 é um X inverso), as linhas 70 e 80 fazem o movimento do míssil e as linhas de 110 a 180 são detalhes gerais. O que nos interessa está contido no fim da linha 70 e nas linhas 90 e 100. O fim da linha 70 contém **AT A,B+3;**



que posiciona a variável do sistema que indica o endereço do próximo PRINT na memória do vídeo. Essa variável está, no Sinclair, nos endereços 16398 e 16399. O conteúdo do endereço que ela aponta é testado nas linhas 90 e 100. Se o número encontrado for 61 (código de "X" — barreira) ou 8 (código de "█" — fim de tela), será dado o desvio previsto. Alguns micros possuem funções especiais que determinam condições de posições de tela. O uso de variáveis do sistema ou funções especiais depende muito do micro, sendo necessário um estudo mais detalhado das suas características para viabilizar o que foi mostrado.

Para ilustrar o outro método, dê BREAK, em seguida elimine as linhas 90 e 100 e o fim da linha 70, acrescentando:

```
1 DIM X(3)
2 LET X(1)=10
3 LET X(2)=11
4 LET X(3)=12
5 LET Y=30
90 IF (A=X(1) OR A=X(2) OR A=X(3)) AND B+2=Y THEN GOTO 120
100 IF B+1=Y THEN GOTO 150
```

As linhas de 1 a 5 definem o buffer que usamos para guardar as posições da barreira (o vetor X guarda as linhas e a variável Y, a coluna). A linha 90 testa a posição para a qual o míssil vai prosseguir, comparando-a com a posição da barreira. A linha 100 não precisa testar se as posições não são da barreira para detectar o erro do alvo, pois caso fossem, o processamento já teria sido desviado pela linha 90. Se a barreira fosse móvel, seu buffer deveria ser alterado a cada movimento. A grande diferença entre os métodos reside no fato de que o segundo não se importa com o que há na tela.

O uso de um ou outro método varia muito, porém na maioria das vezes, o segundo é mais lento. E um modo de se acelerar o primeiro caso é fazer só um teste na linha 90, eliminando a linha 100. Esse teste verificaria se existe algo na posição desejada, e, em caso positivo, desviaria o processamento, onde então se saberia que tipo de colisão havia ocorrido, com acerto ou não. A velocidade aumentaria, pois o teste a cada passo do míssil seria um, ao invés de dois.

### CONTADORES GERAIS

Os contadores são elementos comuns a praticamente todos os tipos de jogos existentes. Eles representam o balanço geral da situação do jogador num dado momento, pois contam os seus pontos marcados, vidas restantes, número de inimigos eventualmente atingidos, tempo decorrido ou restante, recordes anteriores e muitos outros parâmetros vitais para o desenvolvimento do jogo.

A técnica em si não traz nada de complicado. Na verdade, um contador pode ser um simples incremento em uma variável do tipo LET C = C + 1, onde C representa a expressão matemática de alguma situação real na tela. Eventualmente, poderíamos ter decremento na variável (no caso da perda de uma vida, por exemplo); ou ainda, o número a somar ou subtrair poderia ser maior ou menor que 1 (por exemplo, se determinado alvo valesse 10 pontos, outro 20, etc.).

Usando o último programa visto, poderemos mostrar alguns exemplos de contadores, se incluirmos as seguintes instruções:

```
6 LET L=0
7 LET P=0
45 LET L=L+1
125 LET P=P+1
175 IF L=5 THEN GOTO 190
190 PRINT AT 10,12;"FIM",TAB 2;"O MISSIL ACERTOU ";P;" VEZES"
```

Agora temos dois contadores no programa. Rode-o e veja a diferença. Observe que ele só vai fazer cinco lançamentos (devido ao contador da variável L), que é o número de mísseis. Além disso, o programa dirá quantas barreiras foram atingidas (através do contador da variável P), que é o número de acertos.

As linhas 6 e 7 definem os contadores, as linhas 45 e 125 os incrementam e a 175 faz o teste de fim de mísseis. Perceba que na linha 100, quando comparamos B + 1 com Y, estamos usando um contador (B + 1) que, quando chega ao limite (Y), ocasiona uma ordem de desvio.

Obviamente, o programa mostrado foi feito apenas para mostrar onde se localizam e como são feitas certas rotinas elementares. Para um jogo ser realmente bom, ainda deve contar com outros aspectos.

Fazer um bom programa implica não só no uso puramente computacional de instruções, mas também em técnicas gerais de aperfeiçoamento. Essas técnicas fogem ao estilo "receita de bolo", pois variam muito de acordo com o que se pretende; e é nisso que entra a subjetividade do programador ou a sua arte. Tais aspectos dão a diferença básica entre um programa e outro que, à primeira vista, fazem a mesma coisa. Agora, veremos Recursos Adicionais e Sofisticações.

### VELOCIDADE

Mais do que um recurso, a rapidez é necessária para proporcionar uma ação-reação estimulante para o jogador. Existem micros mais lentos que tornam quase impossível a criação de bons jogos, por torná-los monótonos, notadamente quando forem usadas linguagens de alto nível como, por exemplo, o BASIC. Se pudermos utilizar o BASIC compilado, teremos melhores resultados. Porém, se não dispusermos deste recurso, torna-se quase obrigatório o uso da linguagem de máquina. Algo neste sentido pode ser visto no artigo "Jogue em Ritmo Assembler" (MS n.º 40). Outra técnica consiste em evitar certos procedimentos mais lentos por natureza. Uma discussão mais prolongada sobre este e outros aspectos se encontra nos artigos "Anime-se e Faça Bons Programas em BASIC" (MS n.º 23) e "Otimize seus Programas em BASIC" (MS n.º 31).

Nas rotinas apresentadas, tentamos discorrer sobre a morosidade de algumas e a vantagem de alternativas mais velozes e igualmente eficientes.

### MEMÓRIA DISPONÍVEL

Embora não seja um recurso por si só, uma maior disponibilidade de memória possibilita um maior número de situações.

Por outro lado, a economia de memória é, muitas vezes, um fator limitador no desenvolvimento de jogos. O melhor exemplo disso são os adventures: quanto menor a capacidade de armazenamento, menos situações se tornam possíveis. Entretanto, como no caso da velocidade, existem determinados métodos que levam a um menor consumo de memória. Os principais seriam: evitar comentários, espaços em branco e instruções dispensáveis e usar strings em vez de números. Porém, muito mais pode ser feito. Veja o artigo "Utilização da memória de seu Sistema de Modo Mais Eficiente" (MS n.º 5), para mais dicas.

### COMODIDADE

Um jogo deve ser confortável. Certas vezes, pode ser interessante sacrificar a existência de algum comando para não comprometer a facilidade de ação por parte do jogador. Exemplificando: imagine um jogo (não é fato raro...) que use os comandos básicos de um joystick para movimentos e tiro, porém que necessite, por exemplo, de um comando especial de locomoção (caso típico do hiperespaço do "Asteroids"). Ora, é terrivelmente incômodo abrir mão do joystick para pressionar uma tecla do micro. Ideal seria se todos os comandos pudessem ser acionados no primeiro. Em linguagem de máquina pode-se, com certa facilidade, comparar o pressionamento de mais de uma tecla ao mesmo tempo. Uma opção, então, seria usar o botão de tiro e mais um comando do bastão para ordens especiais, como a citada.

Se os comandos forem do teclado, há que se tomar cuida-



## Quem tem o UP-1237 da Parks não tem um modem. Tem quatro.

O Modem UP-1237 é o quatro em um da Parks porque opera em transmissões assíncronas a velocidades de 1200/75 bps ou 300 bps, no padrão CCITT ou no padrão Bell.

Você vai encontrar nesse modem outros diferenciais que a Parks desenvolveu para oferecer o máximo em tecnologia:

- dispositivo de resposta automática incorporado, possibilitando a formação de redes de computadores usando comunicação via linhas telefônicas discadas.
- comutação manual modem/telefone a partir de

chave no painel frontal, permitindo acesso aos grandes bancos de dados nacionais e internacionais.

- seleção de velocidade no painel frontal.
- facilidade de teste de enlace analógico local para diagnóstico do equipamento.
- visualização da situação de configuração e dos circuitos de interface por meio de onze indicadores luminosos no painel frontal.

Como você sabe, para a Parks o melhor é a perfeição. Por isso, você terá a segurança de um rigoroso controle de qualidade e garantia de 2 anos. Comprove.

**MATRIZ:** Porto Alegre - RS - Av. Paraná, 2335 - fone (0512) 42-5500 - Tlx. (051) 1043 - **FILIAIS:** São Paulo - fone (011) 572-7171 • Curitiba - fone (041) 232-1814 • Rio de Janeiro - fone (021) 240-7443 • Recife - fone (081) 325-2123

**REPRESENTANTES:** Brasília - fone (061) 272-1825 • Salvador - fone (071) 241-3091 • Belo Horizonte - fone (031) 201-5800 • Vitória - fone (027) 227-3122 • Blumenau - fone (0473) 22-1700 • Uberlândia - fone (034) 234-4600 • Cuiabá - fone (065) 321-5020 • Campo Grande - fone (067) 383-5331

**Parks**  
INFORMÁTICA

do com a disposição das teclas, buscando um maior conforto. Elas não devem ser adjacentes (para não as confundirmos), nem muito distantes, e deve-se procurar distribuir as mãos no teclado de forma a não sobrecarregar a tarefa de uma delas.

COMPLEXIDADE

Para jogos de muita ação, é preferível que haja até uma certa simplicidade na idéia básica, e comandos “curtos e grossos”. Como exemplo clássico temos os “Defenders” (usualmente com apenas movimentos verticais e de tiro). Por outro lado, jogos que impliquem maior raciocínio podem ser mais complexos, e alguns chegam mesmo a necessitar disso como fator de semelhança a realidade. É o caso de simuladores-de-vôo, que dispõem de inúmeros comandos, e tendo nisto um dos seus maiores atrativos.

INCREMENTOS GERAIS

Um jogo ganha muito em emoção quando traz em si um sentido de competição; mesmo que esta seja dirigida contra nós mesmos (por exemplo, ao procurarmos bater nossos próprios recordes). A simples existência de um *High Score* na tela aumenta o interesse do jogador. Neste sentido, ainda podemos contar com outros recursos; bônus a cada etapa concluída com êxito; tempo extra em caso de realização de alguma tarefa difícil; elogios para os grandes feitos ou até frases lamentando más atuações.

Quanto mais difícil for alcançar um objetivo, maior interesse temos em atingi-lo. É óbvio que existem limites humanos que não podem ser ultrapassados; logo, ao se estipular, por exemplo, o número de pontos de um jogo que se necessita para ganhar tempo extra, deve haver a preocupação que estes sejam humanamente possíveis de se alcançar.

SOM

O som é, sem dúvida, um importante fator no que diz respeito à integração (envolvimento) do jogador com o que ocorre na tela. De fato, o som se presta, em primeira instância, a dar maior veracidade à simulação de uma situação qualquer no jogo, como o tiro de um canhão.

Existe, porém, um problema operacional no uso do som de forma simples, já que, sendo o processamento das instruções sequencial, não é possível que o tiro de canhão — para usar o exemplo anterior — seja visto na tela exatamente ao mesmo tempo que é ouvido seu ruído. A solução, obviamente, passa pela necessidade de se diminuir ao máximo a diferença entre uma ação e sua audição. O problema, então, fica tão mais patente, quanto mais lento o equipamento ou a linguagem usada. Instruções próximas (se possível, subsequentes) são desejáveis para criar um efeito de simultaneidade.

Outro problema é que os comandos BASIC de geração de som são, via de regra, apenas de notas musicais e não sintetizadores de voz ou geradores de ruído. Habilmente, porém, pode-se fazer uso de notas mais baixas (menores frequências) como ruídos ou de notas intercaladas com silêncio, em ritmo acelerado, para simular helicópteros, metralhadoras, etc.

Um uso mais simples e eficiente do som é como detalhe para indicações gerais: recordes batidos, início ou fim de jogo, troca de naves e outras, justamente por não requisitarem tanto a simultaneidade já citada.

Em equipamentos sem som pode ser usado um recurso ilusório, que é o de escrever na tela o ruído do que se quer ressaltar, próximo ao agente da ação. Apesar de parecer sem graça, à primeira vista, alguns BUMs e BANGs bem localizados durante um jogo empolgante podem conseguir bons resultados.

COR

Diferentemente do som, que pode ser simulado nos micros

mudos (até por rotinas especiais), a cor não admite meio termo: ou se tem ou não; e em caso de existência, seus comandos são razoavelmente fáceis de se compreender, sendo necessário apenas usar um pouco de bom senso na escolha das cores, sem exagero. Como exemplo de abuso, poderíamos citar os primeiros programas de TV coloridos, que, ávidos de demonstrar o novo recurso, desenvolviam cenários de gosto duvidoso com todas as cores do espectro misturadas em cada quadro.

Para dispor de alguma alternativa visual, os micros sem cor podem tentar uma simulação de “cinza”, como aqueles caracteres do ZX81, alternando pontos pretos e brancos (GRAPHICS+SHIFT+A ou H), recurso esse que pode ser ampliado se dispusermos de alta resolução.

RESOLUÇÃO GRÁFICA

Como a cor, este é um recurso de tela, sendo porém mais útil que o primeiro em certos casos. É a velha história: como representar um sapo ou uma galinha com um caráter apenas? Se tivermos alta resolução (e facilidade com sua operação, o que nem sempre ocorre), poderemos então desenhar nosso bicho, ponto a ponto. Sem este recurso, teremos que nos esforçar para usar, dentro do conjunto de caracteres do micro, aqueles que mais convierem ao que se tenta representar. Caímos, assim, outra vez no uso do melhor recurso: nosso bom senso. Um asterisco representando uma aranha não é tão mal assim, já uma letra “A” é lamentável...

Som, cor e resolução gráfica, em verdade, caminham juntos. Com o surgimento dos jogos “3D”, como o Zaxxon — para citar um dos mais famosos — multicoloridos, com mil sons e ótima resolução gráfica, talvez não seja exagero ver um certo romantismo-saudosismo naqueles primeiros videogames, com o Space Invaders original. Mas daqueles bichinhos em duas dimensões e preto e branco tiramos uma grande lição: quando se tem boas idéias e se usa a cabeça, os recursos não se tornam indispensáveis.

CONCLUSÕES

Tudo o que vimos não tem a pretensão de ensinar ninguém a ser um mestre na arte de fazer jogos. São apenas algumas técnicas e refinamentos que julgamos úteis, e, às vezes, indispensáveis. Contudo, aquele toque final, que vai dar a nossa marca registrada, depende só da nossa criatividade, que representa o que sentimos e pensamos a nível individual, posto que somos distintos em idéias e gostos.

Usar as rotinas apresentadas, empilhando-as em camadas estanques, seria pouco criativo e resultaria num jogo ruim. Todas as partes de um programa deste tipo devem interagir. Imagine um avião voando, a uma velocidade X qualquer, e um canhão embaixo, esperando o momento mais adequado para o tiro. Não tem o menor sentido haver um “freio” no avião quando atiramos, por causa do maior tempo que precisamos para desenhar o tiro. Esses outros detalhes devem ser levados em conta quando fizermos um jogo. Ai então podemos dizer: “mas em BASIC não dá!”. E muitas vezes não dá mesmo; e temos que recorrer ao Assembler. Mas, não fiquemos nos culpando. Afinal, as linguagens de alto nível foram criadas para, justamente, termos melhores condições de programar. Se elas são, por vezes, muito lentas e nos obrigam a voltar às linguagens primitivas, a culpa não é nossa...

*Giorgiacomo Ponzo Neto cursa Engenharia Elétrica na UFRJ, onde vem ganhando experiência em programação FORTRAN. Trabalha também em BASIC e Assembler no seu micro da linha Sinclair e utiliza ainda calculadoras programáveis.*



# Sofiline

## O SOFTWARE PARA O SEU MICRO

**F = pode ser gravado em fita K7**  
**J = uso obrigatório de joystick**  
**48 = requer 48K de RAM**  
**64 = requer 64K de RAM**

**JOGOS - APPLE II**  
- Cr\$ 70.000 -  
Sabotage (F) - evite os pára-quadistas  
Sargon (F) - jogo de xadrez  
Draw Poker (F) - jogo de poker  
Blackjack (F) - jogo de 21 (cartas)  
Checkers (F) - jogo de damas  
Fast Gammon (F) - jogo de gamão  
Othello (F) - jogo de tabuleiro  
Lode Runner - pegue os tesouros  
Olympic Decathlon - jogos olímpicos  
Space Raiders (F) - de tipo Centopéia  
The Eliminator (F) - combate espacial  
Night Crawler (F) - enfrente as Centopéias  
Space Eggs (F/J) - caçar ovos espaciais  
Hard Hat Mack - perigos na construção  
Head On (F) - jogo de corrida  
Choplifter (F/J) - resgate com helicóptero  
Sea Dragon - pilote o submarino  
Spy's Demise (F) - jogo de agente secreto  
Star Blazer - combate aéreo  
Viper (F) - alimente a cobra  
Cavern Creatures - penetre na caverna  
Moon Patrol (F) - patrulha lunar  
Ms. Pac Man (F) - a namorada do Pac Man  
Mario Bros (F) - Mário contra o gorila  
Buck Rogers - pilote sua nave em "Zoom"  
Pitfall II - aventuras nas cavernas perdidas  
Castle Wolfenstein - engane os nazistas  
Beyond Castle Wolfenstein - mate Hitler  
Aztec - decifre o enigma da pirâmide azteca  
Stellar 7 - combate no espaço em 3D  
Ribbit (F) - jogo tipo Frogger  
Super Bury - combate as pragas  
Allen Ambush (F/J) - jogo tipo Arcade  
Out Post (F) - defesa a estação espacial  
Defender (F/J) - Defender original  
Dog Fight II (F) - confronto de jatos  
Star Trek - Jornada nas Estrelas  
Jellyfish - pilote um submarino  
Swashbuckler II - duelo de espadachins  
Temple Asphal - aventura no templo  
Flight Simulator (F) - simulador de vôo  
One on One (J) - basquete com 2 jogadores  
- Cr\$ 150.000 -  
Karateka - perfeita simulação de karatê.  
Enfrente vários adversários e liberte a princesa do Akuma.  
Conan - localize e destrua Volta em várias fases e com diferentes obstáculos.  
Ghostbusters (J) - monte uma empresa de caça aos fantasmas e comece sua aventura. Baseado no filme.  
Summer Games (64) - 8 modalidades completas de atletismo, com as mais emocionantes provas olímpicas de verão.

**JOGOS - CP-300/500**  
- Cr\$ 280.000 -  
Flight Simulator (F) - simulador de vôo  
Olympic Decathlon (F) - jogos olímpicos

**JOGOS - APPLE II**  
- Cr\$ 70.000 -  
Flight Simulator (F) - simulador de vôo  
Olympic Decathlon (F) - jogos olímpicos

**JOGOS - CP-300/500**  
- Cr\$ 390.000 -  
Creator - gerador de programas BASIC  
Visicalc - planilha de cálculos  
Scriptit (F) - processador de textos  
Clone III - copiador rápido de discos  
ACCEL 3/4 - compilador BASIC  
Bascom - compilador BASIC  
- Cr\$ 490.000 -  
NEWDOS/80 2.0 - sistema operacional  
DOSPLUS 3.5 - sistema operacional  
Arranger II - arquivo de diretórios  
Profile III - gerador de banco de dados  
SuperSCRIPT - processador de textos  
ZEUS - supereditor assembler  
DisnDATA - super disassembler  
- Cr\$ 500.000 -  
Producer - gerador de programas  
Super Utility 3.2 - superutilitário

**JOGOS - SPECTRUM/TK 90X**  
- Cr\$ 60.000 -  
Fighter Pilot (48) - pilote um F-15 Eagle  
Flight Simulation (48) - simulador de vôo  
Hungry Horace - divertido Pac-Man  
Horace & Spiders - enfrente as aranhas  
Horace Skilling (48) - esque montanha abaixo  
Meteors - destrua os meteoros e naves  
Combate Zone - batalha de tanques em 3D  
TerrorDakti 4D (48) - ação na terra perdida  
Astro Blaster - defesa o Império  
Trader (48) - comércio e ação no espaço  
Styx (48) - ande o labirinto e mate a bruxa  
Tutankhamun (48) - explore a tumba  
Arcadia - sobreviva na travessia fatal  
Space Intruder - jogo tipo Invaders  
Tornado (48) - pilote o avião. Cenário em 3D  
Planetoids - aventura espacial  
Match Point (48) - fantástico tênis em 3D  
Cavelon (48) - uma aventura no castelo  
Galaxy Attack (48) - várias fases no espaço  
Chess - jogo de xadrez  
Android 2 (48) - mate os robôs. Cenário em 3D  
Escape - saia do labirinto sem ser morto.  
Lab 3D - será que você sai dessa?  
Ground Attack - jogo tipo Penetrator  
Delta Wing (48) - pilote um caça F-16 em 3D  
Manic Miner (48) - recolha os objetos da mina  
Potty Painter (48) - preencha os quadrados  
Pool - fantástico jogo de sinuca

**UTILITÁRIOS/APLICATIVOS - APPLE II**  
- Cr\$ 320.000 -  
Visicalc - planilha de cálculos  
Visiplot/Visitrend - gerador de gráficos  
Visidex - organizador de atividades  
Visifile - gerador de banco de dados  
Visichedule - análise financeira PERT/CPM  
Apple LOGO - compilador  
Apple FORTRAN (CP/M) - compilador  
Apple Mechanic - utilitário gráfico  
GraFORTH - compilador  
Locksmith 5.0/F - utilitário de cópia  
Nibbles Away I/UC3 - utilitário de cópia  
The Graphics Magicians - animador gráfico  
The Graphics Solution - editor de gráficos  
Print Shop - gerador de impressos gráficos  
- Cr\$ 450.000 -  
Turbo PASCAL (CP/M) - compilador  
Take 1 - superanimador gráfico

**UTILITÁRIOS/APLICATIVOS - SPECTRUM/TK 90K**  
- Cr\$ 70.000 -  
VU 3D - utilitário gráfico. Telas em 3D  
VU Calc - planilha de cálculos  
VU File - arquivo de dados para uso geral  
Ultra Violet - super disassembler  
MCODER II - compilador BASIC  
Paint Box - editor de telas gráficas  
- Cr\$ 90.000 -  
SoftCALC - planilha de cálculos com GRÁFICOS  
PASCAL 1.6 - compilador  
SpecFORTH - compilador  
ASSEMBLER - supereditor assembler  
Tasword II - processador de texto com 64 colunas

**ARMED PATROL (F)** - patrulha de tanques  
**Allen Defense (F)** - proteja-se dos invasores  
**Meteor Mission (F)** - caçada espacial  
**Outhouse (F)** - proteja o seu banheiro  
**Robot Attack (F)** - destrua os robôs  
**Sea Dragon (F)** - pilote um submarino  
**Missile Attack (F)** - defenda sua cidade  
**Scarfman (F)** - jogo tipo Pac Man  
**Cavern (F)** - escape vivo da caverna  
**Penetrator (F)** - aprofunde-se na cavernas  
**Assault (F)** - fuja dos ágeis inimigos  
**Demon Seed (F)** - acabe com os enormes pássaros  
**Cosmic Fighter (F)** - jogo tipo Invasores  
**Panik (F)** - escape dos robôs  
**Crazy Painter (F)** - pinte a tela, se puder!  
**Chicken** - ajude a galinha  
**Galaxy Invasion (F)** - tipo Invasores  
**Sargon (F)** - jogo de xadrez  
**Asilo (F)** - aventura gráfica  
**Zork I** - superaventura  
**Adventur** - 4 aventuras diferentes  
**FucFuc** - programa pornográfico  
**Ghosts (F)** - cace os fantasmas  
**Rally** - pegue todas as bandeiras durante um rally

**Pfíñere (48)** - recolha as peças do computador  
**Piromania (48)** - apague os incêndios  
**The Pyramid (48)** - aventura na pirâmide  
**3D Tank Duel (48)** - superbatalha de tanques  
**3D Starstrike (48)** - guerra nas estrelas  
**Son of Blagger (48)** - recolha os objetos  
**Psytron (48)** - defenda sua base lunar, em 3D  
**Wheeler (48)** - pilote uma moto no subterrâneo  
**Fall Guy (48)** - baseado no filme DURO NA QUEDA  
**Booga-Boo (48)** - ajude o sapo  
**Arventure (48)** - mate o Conde Drácula  
**Hunçback (48)** - ajude o corcunda  
**Harrier Attack** - combate nas Malvinas  
**SPY v SPY (48)** - guerra dos espies  
**Knight Lore (48)** - aventura no castelo, em 3D  
**Fred (48)** - desvende os perigos da pirâmide  
**Fantastic Voyage (48)** - entre no corpo humano  
**Airwolf (48)** - baseado no seriado ÁGUA DE FOGO  
**1994 (48)** - uma missão no futuro  
**River Raid (48)** - percorra de avião um rio perigoso  
**Blade Alley (48)** - desafio espacial em 3D  
**Maziacs (48)** - cace um tesouro no labirinto  
**3D Ant Attack (48)** - salve a mocinha das formigas  
**Spectres** - um jogo diferente tipo come-come  
**New Poker (48)** - clássico jogo de poker  
**Androids (48)** - aventura no labirinto  
**Chese Cake (48)** - salve os humanos  
**PSSST** - salve sua planta das pragas  
**Cookie** - ajude o cozinheiro  
**Scubadive (48)** - recolha as pérolas do mar  
**Penetrator (48)** - destrua o reator  
**Jetset Willy (48)** - recolha os objetos  
**Tranz Am** - pegue os troféus nos EUA  
**Full Throttle (48)** - corrida de motos, em 3D  
**Chequered Flag (48)** - mantenha a POLE POSITION  
**Worse at Sea (48)** - mantenha o navio flutuando  
**Trashman (48)** - ajude a manter a cidade limpa  
**JetPac** - construa os foguetes  
**Seiddad Attack** - combata os Invasores  
**Aquaplane (48)** - uma lancha num mar de perigos  
**High Moon** - duelo no velho oeste  
**Invasion Force** - destrua a nave mãe  
**Jump Hallager (48)** - pule os obstáculos  
**Night Gunner (48)** - pilote um bombardeio  
**Sir Lancelot** - pegue os objetos no castelo  
**Ghostbusters (48)** - caça aos fantasmas  
**Blind Alley (48)** - baseado no filme TRON  
**Molar Maul** - defenda seus dentes das cáries  
**Galaxians** - destrua as abelhas assassinas  
**Monster in Hell** - cuidado com o inferno

Desejo receber os seguintes programas pelos quais estou enviando um cheque nominal à ATI Editora Ltda., Av. Pres. Wilson, 165 - Grupo 1210 - Centro, CEP 20.030 Rio de Janeiro - RJ. Despesas de correio já incluídas.

Computador: ( ) CP-300/CP-500 ( ) APPLE II ( ) SPECTRUM/TK 90X.	ITEM (s) _____	PREÇO Cr\$ _____
Com:.... K RAM. Desejo os programas em: ( ) FITA ( ) DISKETTE.	_____	_____
NOME: _____	_____	_____
END.: _____	CIDADE: _____	_____
UF: _____	CEP.: _____	TEL.: _____
		TOTAL Cr\$ _____



Aqui o jogo é liberado. Escolha o número, faça suas apostas e divirta-se com este programa para a linha TRS-80.

# Roleta

Antônio José Ávila da Silva

Este jogo, que utiliza 13 Kb de memória RAM, foi desenvolvido em um DGT-100, compatível com o TRS-80. Por ser de domínio popular, ele é muito fácil de ser jogado, mesmo por aqueles que não o conhecem. Os códigos adotados estão explicados no próprio programa, e sempre que quiser revê-los tecle (R). Veja agora algumas dicas:

- Em algumas linhas deste programa, que utilizam o comando PRINT, deve-se adotar o comando (↓), para localização de impressão na próxima linha. A linha 160 é um exemplo.
- Linhas que possuam mais de 240 caracteres devem ser editadas após o 240º caráter para a inclusão dos restantes. Este

é o caso, por exemplo, da linha 430.

- Na linha 370, responsável pelo comando da roleta, se introduzir uma rotina de som para facilitar o jogo, n que esta não se assemelhe com o ruído produzido pela bc ao girar na roleta.
- 370 OUT 255,0:OUT 255,1 :?@602,F\$(C);: FOR . .
- Todos os comandos PRINT deste programa foram es com ?.

Antônio José Ávila da Silva é Engenheiro Civil, possui um DGT- dois anos e trabalha atualmente na Direção Geral do INAMPS.

## Roleta

```

0  'ROLETA II      REVISAO 7/84  A.J.AVILA
100 CLS:CLR100:RANDOM1DIMF$(37),9(21),R(21),S(21),G$(21),X(21)
1A=2000:F=1:PRINTCHR$(23);9478,"R O L E T A":FORAA=1TOA:NEXT:CL
S:PRINT220,"INSTRU-OES (B/N)":GOSUB280:IFU$(0)"S",CLS:GOTO200EL
SECLS:PRINT227,"ROLETA"
110 PRINT284,"ABREVIA-OES ADOTADAS:PRINT"PR-PRETO VE-VERMELHO
HE-MENOR HA-MAIOR PA-PAR IM-IMPAR
C1-COLUNA 1 C2-COLUNA 2 C3-COLUNA 3
D1-DUZIA 1 D2-DUZIA 2 D3-DUZIA 3
120 PRINT"PL-PLENO SP-SEMI-PLENO RU-RUA QU-QUADRA ES-ESQ
UICHO
PL-PLENO 1201231261 PL-PAGA 35 VEZES
SP-SEMI-PLENO 1201231261 SP-PAGA 17 VEZES
RU-RUA 1011021031 RU-PAGA 11 VEZES"
130 PRINT" QU-QUADRA 10B111 QU-PAGA 8 VEZE
S
ES-ESQUICHO 1071101 ES-PAGA 5 VEZES
1181211 C1-C2-C3 2 VEZES"
140 PRINT" 1171201 D1-D2-D3 2 VEZE
S
1161191 VE-PR-IM 1 VEZ
HE-HA-PA 1 VEZ"
150 FORAA=1TOA:NEXT:SET(57,19):SET(58,19):SET(61,22):SET(60,22):
SET(66,25):SET(67,25):SET(54,29):SET(55,29):SET(54,41):SET(55,41)
:GOSUB1020
160 PRINT"MENOR IMEI DE 01 A 18
MAIOR IMAI DE 19 A 36
C1 - 1011041 - - 1341
C2 - 1021051 - - 1351
C3 - 1031061 - - 1361
O ZERO E' DA BANCA
170 PRINT:PRINT"***** COMANDOS NO TECLADO *****
*****0 cacife e' de 100 fichas, para adquirir outro t
ecle (K).
180 PRINT"Quantidade de fichas acima de 900 vai para a memoria.
Jogadores (A) (B) (C) (D) (E) (tecle para jogar).
Para direcionar [ "CHR$(93)" "CHR$(94):PRINT"De (1) a (9) par
a apostar e para anular (a) (b) (c) (d) (e)."
190 PRINT"Para girar a roleta (G), para rever as instruções (R)
e para contabilidade (L). Maximo 4 apostas por jogador." :GOS
UB1020:IFF=20RF=4THEN230
200 CLS:PRINT:INPUT"QUANTOS JOGADORES (MAXIMO 5)":A:IFA(1
ORA)5THEN200ELSEPRINT"NOHE DOS PARTICIPANTES COM 4 LETRAS":A=A+1
:B=124:N=251:P=641
210 O=0+1:IF0(A,INPUTA$(0):A$(0)=LEFT$(A$(0),4):GOTO210
220 CLS:FORD=1TOS1K(0)=1:C(0)=100:IFAS(0)="",A$(0)="" :INXTEL
SEAS(0)=A$(0)+"" :A$(0)=LEFT$(A$(0),4):INEXT
230 PRINT200,"
-----
240 PRINT264," | 03 | 06 | 09 | 12 | 15 | 18 | 21 | 24 | 27 | 3
0 | 33 | 36 | |-----|-----|-----|-----|-----|
-----|
250 PRINT2192," | 02 | 05 | 08 | 11 | 14 | 17 | 20 | 23 | 26 |
29 | 32 | 35 | |-----|-----|-----|-----|-----|
-----|

```

```

SEIFB=59THENQ$="SP":U=17:V=10ELSEIFB=64THENQ$="QU":U=17:V=21ELSE
IFB=69THENQ$="SP":U=20:V=21ELSEIFB=74THENQ$="QU":U=20:V=24ELSEIF
B=79THENQ$="SP":U=23:V=24ELSEIFB=84THENQ$="QU":U=23:V=27
400 IFB=89THENQ$="SP":U=26:V=27ELSEIFB=94THENQ$="QU":U=26:V=30EL
SEIFB=99THENQ$="SP":U=29:V=30ELSEIFB=104THENQ$="QU":U=29:V=33ELS
EIFB=109THENQ$="SP":U=32:V=33ELSEIFB=114THENQ$="QU":U=32:V=36ELS
EIFB=119THENQ$="SP":U=35:V=36
490 GOTO680
500 IFB=9THENQ$="PL":U=2:V=2ELSEIFB=14THENQ$="SP":U=2:V=5ELSEIFB
=19THENQ$="PL":U=5:V=5ELSEIFB=24THENQ$="SP":U=5:V=8ELSEIFB=29THE
NQ$="PL":U=8:V=8ELSEIFB=34THENQ$="SP":U=8:V=11ELSEIFB=39THENQ$="
PL":U=11:V=11ELSEIFB=44THENQ$="SP":U=11:V=14
510 IFB=49THENQ$="PL":U=14:V=14ELSEIFB=54THENQ$="SP":U=14:V=17EL
SEIFB=59THENQ$="PL":U=17:V=17ELSEIFB=64THENQ$="SP":U=17:V=20ELSE
IFB=69THENQ$="PL":U=20:V=20ELSEIFB=74THENQ$="SP":U=20:V=23ELSEIF
B=79THENQ$="PL":U=23:V=23ELSEIFB=84THENQ$="SP":U=23:V=26
520 IFB=89THENQ$="PL":U=26:V=26ELSEIFB=94THENQ$="SP":U=26:V=29EL
SEIFB=99THENQ$="PL":U=29:V=29ELSEIFB=104THENQ$="SP":U=29:V=32ELS
EIFB=109THENQ$="PL":U=32:V=32ELSEIFB=114THENQ$="SP":U=32:V=35ELS
EIFB=119THENQ$="PL":U=35:V=35
530 GOTO680
540 IFB=9THENQ$="SP":U=1:V=2ELSEIFB=14THENQ$="QU":U=1:V=5ELSEIFB
=19THENQ$="SP":U=4:V=5ELSEIFB=24THENQ$="QU":U=4:V=8ELSEIFB=29THE
NQ$="SP":U=7:V=8ELSEIFB=34THENQ$="QU":U=7:V=11ELSEIFB=39THENQ$="
SP":U=10:V=11ELSEIFB=44THENQ$="QU":U=10:V=14
550 IFB=49THENQ$="SP":U=13:V=14ELSEIFB=54THENQ$="QU":U=13:V=17EL
SEIFB=59THENQ$="SP":U=16:V=17ELSEIFB=64THENQ$="QU":U=16:V=20ELSE
IFB=69THENQ$="SP":U=19:V=20ELSEIFB=74THENQ$="QU":U=19:V=23ELSEIF
B=79THENQ$="SP":U=22:V=23ELSEIFB=84THENQ$="QU":U=22:V=26
560 IFB=89THENQ$="SP":U=25:V=26ELSEIFB=94THENQ$="QU":U=25:V=29EL
SEIFB=99THENQ$="SP":U=28:V=29ELSEIFB=104THENQ$="QU":U=28:V=32ELS
EIFB=109THENQ$="SP":U=31:V=32ELSEIFB=114THENQ$="QU":U=31:V=35ELS
EIFB=119THENQ$="SP":U=34:V=35
570 GOTO680
580 IFB=9THENQ$="PL":U=1:V=1ELSEIFB=14THENQ$="SP":U=1:V=4ELSEIFB
=19THENQ$="PL":U=4:V=4ELSEIFB=24THENQ$="SP":U=4:V=7ELSEIFB=29THE
NQ$="PL":U=7:V=7ELSEIFB=34THENQ$="SP":U=7:V=10ELSEIFB=39THENQ$="
PL":U=10:V=10ELSEIFB=44THENQ$="SP":U=10:V=13
590 IFB=49THENQ$="PL":U=13:V=13ELSEIFB=54THENQ$="SP":U=13:V=16EL
SEIFB=59THENQ$="PL":U=16:V=16ELSEIFB=64THENQ$="SP":U=16:V=19ELSE
IFB=69THENQ$="PL":U=19:V=19ELSEIFB=74THENQ$="SP":U=19:V=22ELSEIF
B=79THENQ$="PL":U=22:V=22ELSEIFB=84THENQ$="SP":U=22:V=25
600 IFB=89THENQ$="PL":U=25:V=25ELSEIFB=94THENQ$="SP":U=25:V=28EL
SEIFB=99THENQ$="PL":U=28:V=28ELSEIFB=104THENQ$="SP":U=28:V=31EL8
EIFB=109THENQ$="PL":U=31:V=31ELSEIFB=114THENQ$="SP":U=31:V=34ELS
EIFB=119THENQ$="PL":U=34:V=34
610 GOTO680
620 IFB=9THENQ$="RU":U=1:V=3ELSEIFB=14THENQ$="ES":U=1:V=6ELSEIFB
=19THENQ$="RU":U=4:V=6ELSEIFB=24THENQ$="ES":U=4:V=9ELSEIFB=29THE
NQ$="RU":U=7:V=9ELSEIFB=34THENQ$="ES":U=7:V=12ELSEIFB=39THENQ$="
RU":U=10:V=12ELSEIFB=44THENQ$="ES":U=10:V=15
630 IFB=49THENQ$="RU":U=13:V=15ELSEIFB=54THENQ$="ES":U=13:V=18EL
SEIFB=59THENQ$="RU":U=16:V=18ELSEIFB=64THENQ$="ES":U=16:V=21ELSE
IFB=69THENQ$="RU":U=19:V=21ELSEIFB=74THENQ$="ES":U=19:V=24ELSEIF
B=79THENQ$="RU":U=22:V=24ELSEIFB=84THENQ$="ES":U=22:V=27
640 IFB=89THENQ$="RU":U=25:V=27ELSEIFB=94THENQ$="ES":U=25:V=30EL
SEIFB=99THENQ$="RU":U=28:V=30ELSEIFB=104THENQ$="ES":U=28:V=33ELS
EIFB=109THENQ$="RU":U=31:V=33ELSEIFB=114THENQ$="ES":U=31:V=36ELS
EIFB=119THENQ$="RU":U=34:V=36
650 GOTO680
660 IFB=9THENQ$="C1":U=1:V=34:GOTO680ELSEIFB=19THENQ$="C2":U=2:V
=35:GOTO680ELSEIFB=29THENQ$="C3":U=3:V=36:GOTO680ELSEIFB=39THENQ
$="PR":U=0:V=0:GOTO680ELSEIFB=49THENQ$="VE":U=0:V=0:GOTO680ELSEI
FB=59THENQ$="HE":U=1:V=18:GOTO680
670 IFB=69THENQ$="HA":U=19:V=36ELSEIFB=79THENQ$="PA":U=0:V=0ELSE
IFB=89THENQ$="IM":U=0:V=0ELSEIFB=99THENQ$="D3":U=25:V=36ELSEIFB=
109THENQ$="D2":U=13:V=24ELSEIFB=119THENQ$="D1":U=1:V=12ELSE280
680 E=1+1:1+1:IFE121THENF=2:GOTO280ELSEDD=INT(E/4+.75):IFI)AAN
DF(4)THENDD=DD-1:GOTO700
690 IFF=4THENU=0(E):V=R(E):GOTO710ELSEG(E)=Q$=Q$+"":Q(E)=U:R(E)
=U15(E):T
700 IFD(DD)(S(E),S(E)=0:GOTO700ELSEPP=64+X(E)=P:PRINTPP,G$(E)"
":GOSUB720:C(DD)=C(DD)-S(E):PP=640:IFE=17ORE=17ORE=20THEN
PRINTS(E):GOTO750ELSEPRINTS(E)":IFE=17:GOTO750
710 IFX(E)=0ORS(E)=0THEN680ELSEIFE=17ORE=17ORE=20THENPRINT
QX(E),G$(E)" ":GOSUB720:PRINTS(E):GOTO680ELSEPRINTQX(E),G$(E)"
":GOSUB720:PRINTS(E)":I=1:GOTO680
720 IFU(10THENPRINTUSING"0M":U:ELSEPRINTUSING"MM":U:
730 IFV(10THENPRINTUSING"0M":V:ELSEPRINTUSING"MM":V:
740 RETURN
750 D=D+1:PRINTAPP,K(D):A$(D):IFC(D)<0THENPRINT" 000":ELSEIFC(
D)<10THENPRINTUSING" 00M":C(D):ELSEIFC(D)<100THENPRINTUSING" 00M
":C(D):ELSEPRINTUSING" MMN":C(D):
760 PP=PP+13:IFD(5THENPRINT" ":GOTO750ELSEDD=0
770 IFF=3THEN820ELSEIFF=4THEN=0:GOTO680ELSE280
780 GOSUB970:PRINT2592,"LIMITE DE APOSTAS DO JOGADOR (A$(DD)):G
OTO280
790 FORU=1TO15:PRINT2100,"":NEXT:PRINT220,"COMPRA DE CACIFE",,
" NA SUA VEZ TECLE (S/N)".10$=" COMPROU 100 FICHAS:1$=" CACIFES
NA MEMORIA":D=0
800 D=D+1:PRINTA$(D):GOSUB280:IFU$(X$),PRINT0$(K(D)=K(D)+1:C(D)
=C(D)+100:IFK(D)>9THENK(D+5)=K(D+5)+B:K(D)=K(D)-B:PRINT "K(D+5)
":R$
810 IFD(4)THENGOSUB1020:D=0:F=4:GOTO230ELSEPRINT:GOTO800
820 PRINT2620,"APURA-O":F=3:L=L+1:H=0+1:IP$=G$(L):R=Q(L):S=R(
L):T=S(L):I1FL:20THEN970
830 IFD(0THEN820ELSEIFF$="PL"ANDR=ZANDS=Z,H=T+35:T:GOTO920ELSEIF
P$="SP"ANDR=ZANDS=Z+30RP$="SP"ANDR=ZANDS=Z+10RP$="SP"ANDR=Z-1AND
S=20RP$="SP"ANDR=Z-3ANDS=Z,H=T+17:T:GOTO920
840 IFP$="RU"ANDR=ZANDS=Z+20RP$="RU"ANDR=Z-1ANDS=Z+10RP$="RU"AND
R=Z-2ANDS=Z,H=T+11:T:GOTO920
850 IFP$="QU"ANDR=ZANDS=Z+40RP$="QU"ANDR=Z-1ANDS=Z+30RP$="QU"AND
R=Z-2ANDS=Z0RP$="QU"ANDR=Z-3ANDS=Z+1THENH=T+B:T:GOTO920
860 IFP$="ES"ANDR=ZANDS=Z+50RP$="ES"ANDR=Z-1ANDS=Z+40RP$="ES"AND
R=Z-2ANDS=Z+30RP$="ES"ANDR=Z-3ANDS=Z+20RP$="ES"ANDR=Z-4ANDS=Z+10
RP$="ES"ANDR=Z-5ANDS=Z,H=T+5:T:GOTO920
870 IFP$="D1"ANDR=1ANDS=12ANDZ=12THENH=T+T:T:GOTO920ELSEIF
P$="D3"ANDR=13ANDS=24ANDZ=13ANDZ=24THENH=T+T:T:GOTO920ELSEIF
P$="D3"ANDR=25ANDS=36ANDZ=25ANDZ=36THENH=T+T:T:GOTO920
880 IFH$="C1"ANDP$="C1",H=T+T:T:GOTO920ELSEIFH$="C2"ANDP$="C2".H

```

Mensagem de erro

- Em MS n° 46, no programa Missão Impossível, pág. 55, a listagem apresentou erros em algumas linhas. Na linha 86, após "PA-RABENS", o correto é haver apenas uma vírgula, enquanto que a linha 88 já deve começar com PRINT. Confira abaixo como ficam as linhas.

86 PRINTCHR\$(23);PRINTTAB(6)"P A R A B E N S I",TAB(6)STRING\$(15,CHR\$(131)):PRINT208,"Seu escore e'",um dos 5 maiores do dia."

88 PRINT265B,"- - - - -"
- Em MS n° 48, na Rodada MS, programa Projeto de filtros passa-faixa, pág. 75, a listagem apresentou incorreções. Na linha 1970, o M deve ser substituído por N, ficando assim: 1970 PLOT 37 + N,6.

Também ficou faltando a última linha da listagem, ou seja, a li-nha 5100 GOTO 10.
- Em MS n° 50, na reportagem O Mercado Editorial de Infor-mática, pág. 10, no quadro com os livros mais vendidos, saiu erro do endereço da Editora Campus. O endereço certo é: Rua Barão de Itapagipe, 55, CEP 20261, RJ.
- Em MS n° 50, no artigo A função UDG do TK90X, pág. 42, quarto parágrafo, a sexta linha ("... se utiliza de 8 bits."), apresen-tou erro. A frase correta é "... se utiliza de 8 bytes." Já na figura 10, a linha 9010 apresentou falhas. Corrija-a para:

9010 SAVE "garotog"CODE USR "a",

10000 6000- 60 00 C2 00 C4 00 C3 00
- Em MS n° 50, no programa Maçã ou Maca, pág. 36, listagem 1, a linha 6000 saiu com erro; no lugar de 0F, deve-se usar 60. A linha fica assim:

6000- 60 00 C2 00 C4 00 C3 00
- Em MS n° 50, Curso de FORTH, pág. 60, no programa se-guinte ao décimo quarto parágrafo, o traço (-) depois de l deve ser substituído por um ponto (.). O programa fica assim:

10000 6000- 60 00 C2 00 C4 00 C3 00
- Em MS n° 50, no programa Maçã ou Maca, pág. 34, houve erro no décimo quinto parágrafo. O correto é: "A linha 110 coloca o endereço do início da tabela de formas (24576 em decimal, ou seja, 6000 em hexadecimal);",

Imprimir em quatro direções ou armazenar uma tela na memória do seu micro, compatível com o ZX Spectrum, são apenas alguns dos recursos que você conseguirá com o uso deste Monitor gráfico.

# Monitor gráfico

Divino Carlos R. Leitão

Os usuários da linha ZX Spectrum têm agora um programa para auxiliá-los na criação de telas de apresentação, menus e qualquer tipo de gráfico que utilize apenas os caracteres, inclusive os redefinidos.

Este *Monitor Gráfico* é uma ferramenta que permitirá o uso de alguns recursos gráficos do seu micro, diretamente no vídeo, enquanto você cria uma tela. Alguns dos comandos disponíveis são: **INK**; **PAPER**; **FLASH**; **BRIGHT** e **INVERSE**, sendo que o comando **OVER** também foi implementado, mas apenas para que o efeito possa ser visto no vídeo, pois para armazenar os dados referentes a esse comando nas telas criadas, seria necessário aumentar o tamanho das mesmas em pelo menos 704 bytes e uma das características das telas do monitor é justamente economizar memória. Porém, como você poderá ver, este comando não fará falta.

Você ainda poderá encher o vídeo com um caráter; copiar a tela na impressora; imprimir em quatro direções; inserir e suprimir caracteres; fazer um **SCROLL** para cima e para baixo; e se o seu micro é um TK90X, poderá acessar a função **UDG 2**, diretamente no Monitor.

Uma das vantagens de uma tela criada pelo Monitor, além dos recursos gráficos prontos para serem usados, é que cada tela gasta apenas 1625 bytes e já traz incluída uma rotina de impressão em **Assembler**, tendo ainda um **UDG** próprio, que não afeta o uso do **UDG** presente na memória. As telas criadas por este processo poderão ser alocadas em qualquer posição disponível da RAM e chamadas por **RAND USR**, sendo que a única preocupação do usuário será deixar livre na memória um espaço de 1625 bytes para cada tela que desejar juntar ao seu programa.

Após criar uma tela, grave-a em fita cassete usando a opção 7 do menu. Para recuperar a tela gravada, use o comando **LOAD** "CODE XXXX", onde XXXX é o endereço de início da tela e também será o endereço de chamada para imprimi-la no vídeo.

**OBS.:** Para chamar uma tela diretamente de uma rotina em **Assembler**, o par BC deverá ser carregado com o valor XXXX antes de ser executado o comando **CALL XXXX**, portanto não esqueça de carregar o par BC ou haverá problemas.

Há também a opção de gravar as telas no formato padrão do micro, com a opção 5 do menu, ou ainda gravar e carregar apenas o **UDG**. Com isto, você poderá criar, em fita cassete, um banco com vários **UDGs** diferentes. As opções 1 e 2 do menu servem respectivamente para iniciar a criação de uma tela ou editar a tela que estiver na memória. Cuidado ao usar a

opção 1, pois se houver alguma tela na memória, ela será apagada.

O *Monitor Gráfico* não permite que se trabalhe com telas que não tenham sido criadas por ele, pois usa um buffer próprio de trabalho com um sistema específico de armazenamento distribuído da seguinte forma: os 49 bytes iniciais contêm a rotina de impressão que acompanha cada tela, depois vem 1408 bytes, alternando caráter e atributo, ou seja, primeiro byte = **CHR\$**; segundo byte = atributos do **CHR\$** anterior, e assim sucessivamente até totalizar as 32 colunas e 22 linhas; quanto aos 168 bytes restantes, são reservados para os caracteres do **UDG** próprio que cada tela contém.

Além das rotinas em linguagem de máquina, o programa tem uma parte em **BASIC**, para facilitar o uso das rotinas padrão de gravação e **UDG 2**.

Para digitar o programa, use o "Monitor Assembler no TK90X" (publicado em MS nº 51). Inicialmente, deverá ser criada a linha 1 REM com 4800 bytes para receber os dados em hexadecimal da listagem 1. Você irá reparar que não há 4800 dados nesta listagem, isto porque os bytes restantes serão preenchidos pelo próprio programa ao ser utilizado. Digite os dados com atenção e no final entre com a listagem abaixo, usando-a para conferir o trabalho:

```
100 REM EXECUTE COM -> RUN 100
110 LET S=0
120 FOR F=2359 TO 26633 STEP 2
130 LET S=S+PEEK F-PEEK (F+1)
140 PRINT AT 10,13;F
150 NEXT F
160 IF S=4194 THEN PRINT TAB 2;
"OK! OS DADOS ESTAO CORRETOS."
STOP
170 PRINT "FLASH 1: TAB 5; "HA"
ERRO DE DIGITACAO." TAB 5; "CONF"
RA OS DADOS..." STOP
```

Se o resultado for negativo, use o "Monitor Assembler no TK90X" para conferir a entrada de dados e corrija os erros. Após ter certeza de que o programa foi digitado corretamente, dê o comando direto **POKE 23756,0** para proteger a linha com as rotinas em **Assembler**, e ela então irá se transformar em linha 0 e não poderá mais ser editada. Em seguida, elimine o "Monitor Assembler no TK90X" e as linhas de conferência para então entrar com a listagem 2. Antes de testar o programa, grave tudo com o comando direto: **SAVE "MG" LINE 7**.

**OBS.:** Se o seu micro não é um TK90X, mude a linha 11 para 11 **GOTO 9**.

## COMO USAR O MONITOR

Na figura 1, você tem uma tabela com as teclas de controle do Monitor e o seu significado, sendo que algumas devem ser pressionadas em conjunto com **CAPS SHIFT(CS)** e outras com **SYMBOL SHIFT(SS)**. O cursor indicará qual tipo de impressão está sendo usada. As convenções são:

C — imprime maiúsculas ou minúsculas junto com CS; L — imprime minúsculas ou maiúsculas junto com CS; G — imprime apenas os caracteres gráficos; S — imprime apenas os símbolos gráficos; e U — imprime apenas os caracteres redefinidos.

## ROTINAS DE IMPRESSÃO NO VÍDEO

Se você já programa em linguagem de máquina ou está interessado em aprender, continue a leitura pois a partir de agora serão explicadas algumas das técnicas de impressão no vídeo em **Assembler**.

Quem já utilizava linguagem de máquina nos modelos Sinclair (ZX-81), sabe que para imprimir no D-File basta colocar o código do caráter a imprimir no endereço correspondente à posição desejada do D-File. Nos compatíveis com o ZX Spectrum, a área correspondente ao D-File dos modelos ZX-81, que a partir de agora será chamada apenas de MV (Memória de Vídeo), tem 6144 posições e mais 768 bytes que correspondem à área de atributos, — **ATTR**, para simplificar — portanto a impressão em **Assembler** fica um pouco mais complicada.

Em primeiro lugar, para se usar a MV é necessário abrir antes um canal de comunicação com o processador de vídeo. A maneira mais fácil de fazê-lo é usando uma rotina da ROM. Veja como, no exemplo 1:

```
3E02 LD A,02H ;A=2 QUE E' O CANAL DA MV
CD0116 CALL 1601H ;ABRE O CANAL 2
C9 RET ;RETORNA
```

**OBS.:** O canal 2(02H) corresponde apenas às 22 primeiras linhas de vídeo, de 0 a 21; para se usar as linhas 22 e 23, que chamaremos de linhas de edição, o canal aberto terá que ser 0(00H) ou 1(01H). Você deverá ter bastante cuidado quando desejar imprimir nestas duas áreas do vídeo, pois, apesar de estarem em sequência, o uso das mesmas é bem distinto. Mais à frente, você verá outras explicações sobre as diferenças das mesmas.

Após abrir o canal de vídeo, podemos imprimir direto na MV e a maneira mais prática ainda é usar as rotinas da ROM, mais especificamente o conhecido (para os já iniciados em **Assembler**) **RST 10**. Para quem não conhece, o **RST 10** é um comando de um só byte que dá acesso às rotinas de impressão dos micros Sinclair. Veja o exemplo 2:

```
3E30 LD A,30H ;A=30 QUE E' CODIGO DO CHR$ 0
D7 RST 10H ;IMPRIME O CHR$ CONTIDO EM A
C9 RET ;RETORNA
```

Como pode ser visto, não é tão difícil assim, mas não se alegre antes do tempo, pois, com o que vimos até agora, o máximo que podemos conseguir é imprimir um caráter descontroladamente na MV, sem saber qual é a sua cor, fundo ou mesmo posição onde será impresso. Para controlar estes pequenos detalhes, existem os caracteres de controle, que chamaremos de CTRL, ao todo são 32. Na figura 2, você tem uma tabela com a indicação das funções para as quais estes caracteres podem ser utilizados na impressão.

Os códigos de 0(00H) a 3(03H) devem ser usados seguidos de **CALL 1601H** (veja exemplo 1) para abrir os canais correspondentes, sendo que o canal 3(03H) corresponde à impressora. Já os CTRL de 6(06H) a 23(17H) — com exceção de 7(07H); 14(0EH) e 15(0FH) que não devem ser usados na impressão — são todos usados seguidos de **RST 10**. Veja o exemplo 3:

TECLA	CS	SS	OPERACAO
1	XX		LINHA 0 VIDEO COM UN CHR\$
2	XX		MUDA MODO CURSOR C (-) L
3	XX		INFORMA PARAMETROS DO CURSOR
4	XX		EXECUTA COMANDO COPY
5 a 8	XX		NOVA CURSOR SEA ALTERAR VIDEO
9	XX		MUDA DIRECAO DA IMPRESSAO
10	XX		CURSOR NO MODO 6
11	XX		CURSOR NO MODO 8
12	XX		CURSOR NO MODO S
13	XX		DELETE NORMAL
14	XX		COMPRIME LINHA
15	XX		EXPANDE LINHA
16	XX		VOLTA AO MENU
17	XX		SCROLL SOME
18	XX		SCROLL DESCE
19	XX		ACIONA FUNCOES EM VERMELHO
20	XX		CURSOR NO MODO U (UDG)
21	XX		COLOCA CURSOR NA LINHA SEGUINTE
22	XX		
23	XX		
24 a 31	XX		

Figura 1

```
3E16 LD A,16H ;A=16 QUE E' O CTRL DE PRINT AT
D7 RST 10H ;PREPARA PARA RECEBER PARAMETROS
3E05 LD A,05H ;A=5 QUE E' O NUMERO DA LINHA
D7 RST 10H ;POSICIONA NA LINHA
3E0A LD A,0AH ;A=10 QUE E' O NUMERO DA COLUMNA
D7 RST 10H ;POSICIONA NA COLUMNA
C9 RET ;RETORNA
```

Após executar esta sequência, o computador estará pronto para imprimir na linha 5, coluna 10, o caráter que for ordenado. Atenção, isto só é válido se o canal da MV corresponde às linhas 0 a 21 (exemplo 1) já tiver sido aberto antes.

Para estabelecer os parâmetros de cor, brilho e outros recursos de seu micro, use os CTRL de 16(10H) a 21(15H). Veja o exemplo 4:

```
3E10 LD A,10H ;A=10 QUE E' O CTRL DE INK
D7 RST 10H ;PREPARA PARA RECEBER PARAMETROS
3E07 LD A,07H ;A=7 QUE CORRESPONDE A COR BRANCA
D7 RST 10H ;MUDA INK PARA BRANCO
C9 RET ;RETORNA
```

Para mudar o **PAPER**, executa-se a mesma sequência, sendo que os códigos de cor variam de 0(00H) ou preto, até 7(07H) que é a cor branca, o que corresponde à disposição das cores no teclado. Para mudar os outros parâmetros (**FLASH**; **BRIGHT**; **INVERSE** e **OVER**), usa-se antes do segundo **RST 10** apenas 1(01H) que significa ligado ou 0(00H) que significa desligado. Veja exemplo (5) com o **FLASH**:

```
3E12 LD A,12H ;A=12 QUE E' O CTRL DE FLASH
D7 RST 10H ;PREPARA PARA RECEBER PARAMETROS
3E01 LD A,01H ;A=1 QUE CORRESPONDE A LIGADO
D7 RST 10H ;MUDA FLASH PARA LIGADO
C9 RET ;RETORNA
```

Após executar o último exemplo, o caráter a ser impresso estaria com o **FLASH** ligado, ou seja, ficaria piscando, alterando seu **INK** e **PAPER**.

Os últimos exemplos valem para todos os CTRL que podem ser usados na impressão, basta chamar **RST 10** após o acumulador ter sido carregado com o CTRL desejado e depois repetir a operação tendo sido o acumulador carregado com um código específico da função, o qual pode ser o valor de uma cor, a posição de uma coluna ou o correspondente a ligar e desligar a função. A única exceção é o CTRL de **PRINT AT** que precisa de dois parâmetros após ser acionado, ou seja, linha e coluna, nesta ordem. Como opção para o **PRINT AT**, você pode usar a rotina da ROM a partir de 8202 (200AH), sendo que antes é necessário carregar o par BC com o valor da linha (C) e coluna (B). Veja no exemplo 6, o que acontece se usarmos esta rotina:

```
3E16 LD A,16H ;A=16 QUE E' CTRL DE PRINT AT
D7 RST 10H ;PREPARA PARA RECEBER PARAMETROS
79 LD A,C ;A=VALOR DA LINHA CONTIDO EM C
D7 RST 10H ;POSICIONA LINHA
78 LD A,B ;A=VALOR DA COLUMNA CONTIDO EM B
D7 RST 10H ;POSICIONA COLUMNA
C9 RET ;RETORNA
```



Todos os exemplos mostrados até agora permanecem depois de usados, até que seja executada uma rotina que os modifique, ou seja, se você abrir um canal da MV, ele permanecerá aberto até outro canal ser ativado; como algumas rotinas do sistema podem desativar este canal, é conveniente que ele seja aberto toda vez que se iniciar qualquer rotina de impressão. Quando se imprime um caráter (veja exemplo 2), a posição do cursor avança uma coluna ou, se for a última coluna, volta à coluna 0 e avança uma linha; ao atingir a linha 22 será solicitada o comando **SCROLL**.

Na área de edição acontece algo semelhante, mas as linhas são contadas de baixo para cima, ou seja, a linha 23 passa a ser a linha 0. Uma outra diferença da área de edição é que para se imprimir com segurança, nesta área, é necessário que ela esteja limpa. Se tentarmos a impressão de um caráter sobre outro já existente, será executado um **SCROLL** no vídeo todo. Portanto, use a área de edição com muito cuidado para não perder o controle da impressão.

Para deixar mais claro o uso do CTRL, o exemplo a seguir mostra como se pode encher o vídeo com um caráter, que será o correspondente ao A do UDG, cujo código é 144 (90H). Para cor de fundo (PAPER), será colocado o verde e a cor do caráter (INK) será o preto, que deverá ficar piscando (FLASH = 1), e também não terá muito brilho (BRIGHT = 0), além de não ser invertido (INVERSE = 0) e não ter transparência (OVER = 0). Aqui vale um aparte para os que ainda não compreenderam o comando **OVER**: a sua finalidade é criar prioridade de impressão, um caráter impresso com **OVER = 0** sobrepõe um outro quando for impresso, ocupando todo o espaço; já um caráter com **OVER = 1**, será misturado com o caráter que estiver abaixo dele, criando um novo desenho que corresponderá à mixagem dos dois, dando ainda a ilusão de que o caráter com **OVER = 1** é transparente. Vamos então ao exemplo 7:

3E02 D7	LD CALL	A,02H 1601H	IA=2 QUE E' O CANAL DD MV TÁBRE CANAL 2
010000 CD0A20	BC,00H CALL	BC,00H 200AH	IB=0 (COLUNA) C=0 (LINHA) ROTINA DA ROM QUE POSICIONA PRINT AT ATRAVES DE PARAMETROS DO PAR DE REGISTRADORES BC
3E10 D7	LD RST	A,10H 10H	IA=16 QUE E' O CTRL DE INK PREPARA PARA RECEBER PARAMETROS

## Listagem 1

```
23759 EA 0D 80 21 6A 5C CB 9E = 967
23767 AF 32 86 67 32 8B 67 32 = 897
23775 89 67 32 8A 67 32 8B 67 = 967
23783 32 8E 67 32 8A 67 32 85 = 907
23791 67 32 8F 67 3E 01 32 83 = 739
23799 67 3E 28 32 82 67 3E 20 = 630
23807 32 00 67 3E 38 32 C2 67 = 810
23815 3E 07 32 87 67 3E 20 32 = 549
23823 8D 67 32 A2 65 3E 43 32 = 784
23831 8C 67 21 78 00 22 D0 67 = 789
23839 21 0A 68 22 C6 67 22 C6 = 724
23847 67 21 88 6D 22 C6 67 21 = 757
23855 00 58 22 C4 67 22 C6 67 = 756
23863 21 8F 5A 22 C8 67 21 8A = 822
23871 6D 22 78 5C 21 D6 5A 22 = 729
23879 D2 67 CD F7 63 CD 49 62 = 1240
23887 CD 36 6E 21 56 66 7E FE = 970
23895 03 28 12 FE 02 0A 23 = 394
23903 46 3E 2D 10 F8 23 18 = 705
23911 ED D7 23 18 E9 CD 68 63 = 1152
23919 06 30 90 06 00 4F FE 01 = 538
23927 28 1E FE 02 06 CD 04 = 781
23935 67 C3 B1 5D FE 03 C8 FE = 1279
23943 04 C8 FE 05 C8 FE 06 C8 = 1123
23951 FE 07 C8 FE 08 C8 18 D5 = 1160
23959 CD F7 63 2A C6 67 E8 2A = 1177
23967 C6 67 01 C0 02 3E 20 12 = 608
23975 13 ED 0A 78 81 20 F6 CD = 1196
23983 92 62 CD 83 62 CD 88 62 = 1168
23991 CD 01 64 3A BC 67 FE 55 = 994
23999 28 3E 21 67 C6 CD D8 63 = 862
24007 21 84 65 CD F7 63 CD E2 = 1208
24015 63 3A 88 67 CD 2A 63 CD = 995
24023 E2 63 3A 89 67 CD 2A 63 = 1017
24031 CD E2 63 3A 8A 67 CD 2A = 1124
24039 63 CD E2 63 3A 8B 67 CD = 1182
24047 2A 63 CD E2 63 3E 43 D7 = 1015
```

```
24055 3E 4F D7 3E 4C D7 18 0C = 745
24063 21 1C 66 CD D8 63 21 39 = 776
24071 66 CD CF 63 CD FA 62 3A = 1224
24079 BC 67 FE 55 28 18 21 E0 = 951
24087 5A 3A 86 67 B7 C4 EC 63 = 1147
24095 CD F0 63 23 3A B7 67 B7 = 1106
24103 C4 EC 63 CD F0 63 CD EA = 1514
24111 62 CD 68 63 FE E2 20 06 = 1024
24119 CD 92 62 C3 49 5D FE 06 = 1070
24127 20 13 3A BC 67 FE 43 20 = 753
24135 04 3E 4C 18 02 3E 43 20 = 502
24143 88 62 C3 87 5D FE 0F 20 = 1054
24151 04 3E 47 18 F2 FE 29 20 = 730
24159 04 3E 53 18 EA FE 0E 20 = 707
24167 04 3E 55 18 E2 FE 0C 20 = 699
24175 17 3E 20 32 8D 67 CD 70 = 789
24183 63 CD 74 62 CD 91 61 CD = 1170
24191 83 62 CD 88 62 C3 08 5E = 1019
24199 FE 60 20 1C CD 01 64 21 = 749
24207 2A 65 CD CF 63 CD 68 63 = 1062
24215 21 19 65 CD AC 62 87 28 = 857
24223 F4 23 7E 32 86 67 18 1E = 794
24231 FE 3F 20 2D CD 01 64 21 = 720
24239 40 65 CD CF 63 CD 68 63 = 1092
24247 21 19 65 CD AC 62 87 28 = 857
24255 F4 23 7E 32 87 C3 AE = 1110
24263 5D C3 87 5D FE 2F 20 08 = 908
24271 3A 88 67 CD 6C 61 32 8B = 989
24279 67 1E FE 2A 20 08 3A = 762
24287 89 67 CD 6C 61 32 8B 67 = 1036
24295 18 DF FE 2C 20 08 3A 8A = 832
24303 67 CD 6C 61 32 8A 67 18 = 876
24311 D0 FE 2E 20 08 3A 8B 67 = 899
24319 CD 6C 61 32 8B 67 18 C1 = 967
24327 FE 08 20 0F CD 92 62 CD = 963
24335 91 61 CD 83 62 CD 88 62 = 1166
24343 C3 08 5E FE 0A 20 08 CD = 809
```

```
24351 92 62 CD AD 61 18 E8 FE = 1232
24359 08 20 08 CD 92 62 CD E7 = 936
24367 61 18 DF FE 09 20 08 CD = 852
24375 92 62 CD 74 61 18 D3 FE = 1151
24383 0D 20 15 CD 92 62 CD 74 = 836
24391 61 3A 85 67 B7 20 F7 CD = 1106
24399 83 62 CD 88 62 C3 08 5E = 1019
24407 FE AC 20 26 3A 85 67 FE = 1092
24415 1F CA 2D 5E 47 3E 1F 90 = 680
24423 17 06 00 4F 2A C6 67 23 = 490
24431 09 54 12 28 2B ED 88 CD = 898
24439 E9 63 12 18 3E 20 12 C3 = 684
24447 1D 60 FE 3A 20 0F 3A 83 = 721
24455 67 CD 6C 61 32 83 67 CD = 1050
24463 EA 62 C3 2D 5E FE 07 20 = 959
24471 45 3A C0 67 32 A2 65 CD = 940
24479 01 64 21 8C 65 CD D8 63 = 898
24487 2A D2 67 3A C2 67 77 21 = 862
24495 A7 65 CD CF 63 CD 68 63 = 1187
24503 FE 73 28 07 FE 6E CA C5 = 1179
24511 5E 18 F2 2A CC 67 01 C0 = 902
24519 02 3A C0 67 77 23 3A C2 = 761
24527 67 77 23 08 78 81 20 F1 = 838
24535 CD D4 67 C3 AE 50 FE 28 = 1276
24543 28 0E FE 27 28 0A FE 26 = 689
24551 28 06 FE 25 28 02 18 09 = 412
24559 32 82 67 CD EA 62 C3 2D = 1108
24567 5E FE 5F 20 2D 3A 85 67 = 862
24575 FE 1F CA 2D 5E 47 3E 1F = 790
24583 90 17 06 00 4F 2A C6 67 = 599
24591 54 5D 23 23 ED 00 E8 CD = 1100
24599 E9 63 36 20 23 77 CD D4 = 989
24607 67 CD 83 62 CD 88 62 C3 = 1222
24615 2D 5E FE 4A 20 53 CD 01 = 718
24623 64 21 8A 65 CD D8 63 3A = 1001
24631 8C 67 FE 43 20 05 31 CF = 889
24639 65 18 1E FE 4C 20 05 21 = 555
24647 DA 65 18 15 FE 47 20 05 = 726
24655 21 E5 65 18 0C FE 53 28 = 768
24663 05 21 EF 65 18 03 21 F8 = 686
24671 65 CD D4 63 21 FF 65 CD = 1211
24679 CF 63 3A C0 67 D7 3A C2 = 1126
24687 67 21 F8 5A 77 21 18 66 = 752
```

```
24695 CD D4 63 CD 26 64 C3 B7 = 1237
24703 5D FE C3 20 21 CD E9 63 = 1144
24711 2A CC 67 54 5D 01 40 00 = 591
24719 09 01 40 05 ED 00 06 20 = 530
24727 EB 36 20 23 77 23 10 F9 = 775
24735 CD D4 67 C3 AE 5D FE CD = 1441
24743 20 1E CD E9 63 2A CE 67 = 950
24751 23 54 5D 01 C0 FF 09 01 = 670
24759 40 05 ED 88 66 20 E8 77 = 882
24767 28 36 20 28 10 F9 18 D8 = 677
24775 FE 05 20 16 CD 92 62 3E = 824
24783 03 CD 01 16 CD AC 0E 3E = 684
24791 02 CD 01 16 CD 88 62 C3 = 915
24799 30 5E 3A 8C 67 FE 43 20 = 844
24807 15 21 4A 64 CD A1 62 87 = 875
24815 CA 30 5E 23 7E FE 58 38 = 906
24823 41 28 28 7E 18 3C FE 4C = 691
24831 20 15 21 4A 64 CD A1 62 = 724
24839 B7 CA 30 5E FE 41 38 2A = 944
24847 FE 58 30 26 7E 18 23 FE = 870
24855 47 20 05 21 96 64 18 11 = 432
24863 FE 53 20 05 21 A9 64 18 = 700
24871 08 FE 55 C2 30 5E 21 EC = 952
24879 64 CD AC 62 87 CA 30 5E = 1102
24887 23 7E 32 8D 67 CD 7D 63 = 932
24895 CD 74 62 3A 82 67 FE 25 = 1049
24903 20 05 CD 91 61 18 15 FE = 783
24911 26 20 05 CD AD 61 18 0C = 586
24919 FE 27 20 05 CD E7 61 18 = 887
24927 03 CD 74 61 CD 83 62 CD = 1060
24935 88 62 C3 08 5E 87 28 02 = 810
24943 AF C9 3E 01 C9 3A 85 67 = 982
24951 FE 1F 28 07 3C 32 85 67 = 726
24959 C3 29 62 3A 84 67 FE 15 = 950
24967 CA 49 62 3C 82 84 67 AE = 941
24975 18 E8 3A 85 67 B7 28 07 = 831
24983 3D 32 85 67 C3 39 62 3A = 803
24991 B4 67 B7 CA 5D 62 30 32 = 970
24999 B4 67 3E 1F 18 E8 3A 84 = 873
25007 67 FE 15 28 19 3C 32 84 = 733
25015 67 2A CA 67 01 40 00 09 = 524
25023 22 CA 67 2A C4 67 01 20 = 713
25031 00 09 22 C4 67 C9 AF 32 = 768
25039 B4 67 2A C6 67 3A 85 67 = 974
25047 06 00 4F 09 69 22 CA 67 = 442
25055 2A C6 67 09 22 CA 67 C9 = 886
25063 3A 84 67 B7 28 19 3D 32 = 700
25071 B4 67 2A C6 67 01 C0 FF = 1078
25079 09 22 CA 67 2A C4 67 01 = 690
25087 00 FF 09 22 CA 67 C9 3E = 1084
25095 15 32 84 67 2A CE 67 01 = 706
25103 C2 FF 09 22 CA 85 67 06 00 = 806
25111 4F 09 09 22 CA 67 2A C8 = 678
25119 67 11 E1 FF 19 09 22 CA = 864
25127 67 C9 2A C4 67 23 23 22 = 755
25135 CA 67 2A C4 67 23 22 CA = 911
25143 67 C9 2A C4 67 28 28 22 = 771
25151 CA 67 2A C4 67 28 22 CA = 919
25159 67 C9 2A CC 67 22 CA 67 = 992
25167 2A C6 67 22 CA 67 AF 32 = 901
25175 B4 67 32 85 67 C9 2A CE = 1066
25183 67 22 CA 67 2A C8 67 22 = 821
25191 C4 67 3E 15 32 84 67 3E = 777
25199 1F C2 85 67 C9 3A 8D 67 = 916
25207 2A CA 67 77 23 E8 2A C4 = 974
25215 67 ED 0A C9 2A C4 67 7E = 1174
25223 32 C0 67 2A C4 67 7E 32 = 862
25231 C2 67 C9 CD 8E 63 3A C0 = 1242
25239 67 D7 2A C4 67 3A C2 67 = 1014
25247 77 C9 3A BE 67 BE C8 AF = 1236
25255 B6 C8 23 18 F5 3A BE 67 = 1037
25263 BE C8 AF B6 C8 23 23 18 = 1041
25271 F4 32 BC 67 3A 88 67 32 = 980
25279 8F 67 3E 01 32 88 67 3A = 752
25287 BC 67 32 8D 67 3A 8A 67 = 980
25295 F5 AF 32 8A 67 CD 7D 63 = 1188
25303 F1 32 8A 67 3A 8F 67 32 = 982
25311 88 67 C9 2A D8 67 44 40 = 986
25319 C3 3D 1F 3A 83 67 B7 CA = 1012
25327 E2 62 21 E8 03 11 28 00 = 649
25335 C3 85 03 CD 0C 63 3A 84 = 933
25343 67 CD 2A 63 CD 18 63 3A = 838
25351 85 67 C3 2A 63 3E 00 CD = 887
25359 01 16 3E 16 07 3E 00 D7 = 599
25367 3E 1D 07 C9 3E 00 CD 01 = 775
25375 16 3E 16 07 3E 01 D7 3E = 661
25383 1D 07 C9 26 00 6F 7C 85 = 899
25391 20 07 3E 30 07 3E 2D 07 = 673
25399 C9 11 3E 3A 07 32 8F 67 = 899
25407 AF E8 4E 23 46 23 E8 B7 = 1046
25415 ED 42 38 03 3C 18 F9 09 = 704
25423 B7 20 07 3A 8F 67 B7 28 = 797
25431 E7 AF C6 30 32 8F 67 D7 = 1211
25439 79 FE 01 20 D8 3E 20 D7 = 936
25447 C9 CD 8E 02 0E 00 20 F9 = 845
25455 CD 1E 03 30 F4 55 5F CD = 851
25463 33 03 32 8E 67 C5 CD BE = 993
25471 63 3A 88 67 B7 20 1E 3E = 740
25479 1D 07 3A 86 67 D7 3E 11 = 868
25487 D7 3A 87 67 D7 18 0E 3E = 874
25495 1D 07 3A 87 67 D7 3E 11 = 869
25503 D7 3A 86 67 D7 3E 12 D7 = 1068
25511 3A 88 67 D7 3E 13 D7 3A = 914
25519 89 67 D7 3E 15 D7 3A 8A = 1045
25527 67 D7 3A 8D 67 D7 C9 3E = 1146
25535 02 CD 01 16 3E 16 D7 3A = 587
25543 BA 67 D7 3A 85 67 D7 C9 = 1256
25551 E5 CD 08 64 E1 7E 87 C8 = 1279
25559 D7 23 18 F9 E5 CD 19 64 = 1082
25567 E1 18 F2 3E 20 D7 3E 20 = 894
```

```
25575 D7 C9 3A 87 67 17 17 17 = 829
25583 C9 06 03 77 23 10 FC C9 = 833
25591 3E 02 CD 01 16 06 18 C3 = 517
25599 44 0E 3E 02 CD 01 16 06 = 380
25607 02 C3 44 0E 3E 00 CD 01 = 547
25615 16 3E 16 D7 3E 01 D7 AF = 774
25623 D7 C9 3E 00 CD 01 16 3E = 768
25631 16 D7 AF D7 AF D7 C9 3A = 1276
25639 BE 67 F5 CD 68 63 47 F1 = 1258
25647 88 28 F7 C9 2A 20 54 41 = 895
25655 42 45 4C 41 53 20 2A 10 = 449
25663 27 E8 03 64 00 0A 00 01 = 385
25671 00 00 00 20 20 30 30 31 = 209
25679 31 32 32 33 33 34 34 35 = 408
25687 35 36 36 37 37 38 38 39 = 440
25695 39 61 41 62 42 63 43 64 = 649
25703 44 65 45 66 46 67 47 68 = 688
25711 48 69 49 6A 6A 68 48 6C = 720
25719 4C 6D 40 6E 4E 6F 4F 70 = 752
25727 50 71 51 72 52 73 53 74 = 784
25735 54 75 55 76 56 77 57 78 = 816
25743 58 79 59 7A 5A 7A 00 31 = 681
25751 8E 32 8D 33 C6 3A 88 35 = 768
25759 8A 36 89 37 88 38 8F 20 = 751
25767 20 00 31 21 32 40 33 23 = 314
25775 34 24 35 25 36 26 37 27 = 364
25783 38 28 39 29 30 5F 72 3C = 511
25791 74 3E 79 58 75 5D 6F 38 = 770
25799 70 22 61 7E 73 7C 64 5C = 800
25807 66 78 67 7D 68 5E 6A 2D = 802
25815 68 28 6D 3D 7A 3A 63 3F = 661
25823 76 2F 62 2A 6E 2C 6D 2E = 614
25831 20 20 69 7F 00 61 90 62 = 635
25839 91 63 92 64 93 65 94 66 = 988
25847 95 67 96 68 97 69 98 6A = 1020
25855 99 68 9A 6C 9B 6D 9C 6E = 1052
25863 9D 6F 9E 70 7F 7A 7B 7C = 1084
25871 A1 73 A2 74 A3 75 A4 76 = 1030
25879 20 00 30 00 31 01 32 02 = 182
25887 33 03 34 04 35 05 36 06 = 228
25895 37 07 08 53 65 6C 65 63 = 554
25903 69 6F 6E 65 20 6E 6F 76 = 798
25911 6F 20 49 4E 48 20 20 28 = 473
25919 20 30 20 61 20 37 20 29 = 369
25927 00 53 65 6C 65 63 6F 6F = 708
25935 6E 65 20 6E 6F 76 6F 20 = 725
25943 50 41 50 45 52 20 20 20 = 480
25951 30 20 61 20 37 20 29 00 = 337
25959 49 4E 48 20 50 41 50 20 = 515
25967 46 4C 53 20 42 52 47 20 = 512
25975 4F 56 52 20 49 4E 56 20 = 548
25983 20 4C 49 4E 00 20 20 20 = 355
25991 20 20 20 00 45 4E 43 = 342
25999 48 45 52 20 4F 20 56 49 = 525
26007 44 45 4F 20 43 4F 40 20 = 503
26015 20 3E 20 20 40 3C 20 00 = 308
26023 28 53 29 49 40 20 20 4F = 457
26031 55 20 20 28 4E 29 41 4F = 452
26039 20 3F 00 40 4F 44 4F 20 = 430
26047 44 45 20 49 40 50 52 45 = 550
26055 53 53 41 4F 20 3D 00 40 = 435
26063 40 41 49 55 53 43 55 4C = 611
26071 41 53 00 40 49 4E 55 53 = 544
26079 43 55 4C 41 53 00 20 47 = 479
26087 52 41 50 48 49 43 53 00 = 522
26095 20 20 53 59 40 42 4F 4C = 534
26103 00 20 20 55 44 47 00 32 = 320
26111 43 48 52 24 20 53 4F 42 = 517
26119 20 4F 20 43 55 52 53 4F = 539
26127 52 20 3D 20 20 2D 3E 20 = 378
26135 00 20 3C 2D 00 20 20 41 = 266
26143 42 43 44 45 46 47 48 49 = 556
26151 4A 48 4C 4D 4E 4F 50 51 = 620
26159 52 53 54 55 20 20 4C 49 = 547
26167 4E 00 20 90 91 92 93 = 724
26175 94 95 96 97 98 99 9A 9B = 1212
26183 9C 9D 9E 9F 0A 01 A2 03 = 1276
26191 4A 20 20 43 4F 4C 00 02 = 452
26199 22 8C 8C 8C 8C 8C 20 8D = 843
26207 4F 4E 49 54 4F 52 20 20 = 539
26215 47 52 41 46 49 43 4F 20 = 539
26223 8C 8C 8C 8C 8C 02 0A 8A = 844
26231 02 1A 85 02 0A 8A 02 03 = 310
26239 14 01 31 14 00 20 20 43 = 221
26247 52 49 41 52 20 4E 4F 56 = 577
26255 41 20 54 45 4C 41 02 05 = 398
26263 85 02 0A 8A 02 1A 85 02 = 440
26271 0A 8A 02 03 14 01 32 14 = 238
26279 00 20 20 45 44 49 54 41 = 423
26287 52 20 54 45 4C 41 20 41 = 505
26295 54 55 41 4C 02 03 85 02 = 450
26303 0A 8A 02 1A 85 02 0A 8A = 447
26311 02 03 14 01 33 14 00 20 = 129
26319 20 41 43 49 47 4E 41 52 = 541
26327 20 55 44 47 20 32 02 07 = 347
26335 85 02 0A 8A 02 1A 85 02 = 440
26343 0A 8A 02 03 14 01 34 14 = 240
26351 00 20 20 27 53 41 56 45 = 406
26359 27 20 54 45 4C 41 20 45 = 466
26367 20 55 44 47 02 03 85 02 = 396
26375 0A 8A 02 1A 85 02 0A 8A = 447
26383 02 03 14 01 35 14 00 20 = 131
26391 20 27 53 41 56 45 27 20 = 445
26399 53 43 52 45 45 4E 24 02 = 486
26407 06 85 02 0A 8A 02 1A 85 = 444
26415 02 0A 8A 02 03 14 01 36 = 224
26423 14 00 20 20 27 53 41 56 = 357
26431 45 27 20 55 44 47 02 0A = 376
26439 85 02 0A 8A 02 1A 85 02 = 440
26447 0A 8A 02 03 14 01 37 14 = 243
```

## Listagem 2

```
7)IF PEEK 23675+256*PEEK 2367
6=28042 THEN GOTO 9
8 UDG 0
9 BORDER 7: CLS : LET L=USR 2
3762: IF L<3 AND L<7 THEN RAND U
SR 26585
10 GOTO 8+L
11 CLS : UDG 2: GOTO 9
12 GOSUB 20: GOTO 9
13 GOSUB 30: GOTO 9
14 GOSUB 40: GOTO 9
15 CLS : PRINT "Posicione
a FITA e tecle ENTER": GOSUB 45:
GOSUB 48: PRINT AT 1,6+3*(L=8):
"LOAD" UDG "1"("e TELA" AND L
```



## Magnex lança M-20



Propondo ao usuário que não jogue fora nada do que já gastou em hardware e em software compatibilizando tudo através de um novo sistema multiusuário, a Magnex lançou o M-20. O M-20 é um computador de 8 bits que aceita como terminal qualquer microcomputador, tanto de 8 quanto de 16 bits, desde que compatíveis com CP/M. O novo equipamento da Magnex também emula, ele próprio, terminais Burroughs e Honeywell-Bull. Multiusuário e multitarefa, o M-20 suporta até quatro terminais e duas impressoras, tem 320 Kb de memória e até 20 Mb em disco Winchester. Com uma produção inicial de 30 máquinas/mês, a Magnex espera se consolidar definitivamente no mercado de pequenas e médias empresas, conquistando um aumento substancial no seu parque instalado de computadores. Dependendo das necessidades de cada usuário, o preço do M-20 vai variar de 800 a 1.500 ORTN.

## Promoção Scritta

A Scritta Eletrônica está promovendo junto aos proprietários de micros IBM-PC/XT (e compatíveis) e impressoras Grafix a distribuição gratuita do livro "Conexão Grafix IBM-PC e compatíveis". Se você está interessado em receber o livro, escreva para o De-

partamento de Propaganda da empresa (Alameda Amazonas, 832, Alphaville — Barueri — São Paulo, CEP: 06400) e mencione o modelo do seu microcomputador, o nº da série da impressora e o nome da publicação que veiculou esta informação.

## Inaugurado o Infoshopping

No dia 5 de dezembro o Rio ganhou o primeiro shopping de informática do país. São vinte e seis lojas destinadas à venda de equipamentos ou suprimentos, clubes de vídeo, software-houses, livrarias etc.

No Rio Infoshopping estão empresas conhecidas na área como a Aleph Sistemas e Métodos, Ciência Moderna e Computação, Computer Shopping Moore, Mikros, Racimec, R&T Informática e Unitel.

O shopping conta ainda com dois cinemas, que serão usados, na parte da manhã, para a realização de palestras e seminários. O Rio Infoshopping fica na Rua do Cateite, 311, Largo do Machado (próximo à Estação do Metrô).

## IFPS/PLUS

Através de acordo firmado com a DSS Informática, representante exclusiva da Execucom Systems Corporation no Brasil, a Intertec passou a oferecer a seus clientes o software IFPS/PLUS, atra-

## Apoio ao software

Os que esperavam muitas novidades se frustraram com o 6º Enesi — Encontro Nacional das Empresas de Serviços de Informática, promovido pela Assespro no início de dezembro em São Paulo. O evento estava esvaziado e a exposição paralela de pacotes e equipamentos também não atraiu muita gente. Na abertura do encontro, o Ministro Renato Archer, da Ciência e Tecnologia, celebrou as novas linhas especiais de crédito operantes, como o Proinfo, do BNDES, e principalmente o Programa de Apoio ao Software, da Finep, que instituiu duas modalidades de financiamento — com retorno e com participação no risco — para pequenas software-houses; system-houses e universidades.

vés do seu bureau. Trata-se de um pacote de apoio à decisão, integrado com módulos de otimização e banco de dados relacional. Informações pelo tel.: (011) 259-2055.



## Light pen para TK90X

No final deste mês já estará disponível nas lojas a light pen para se conectar ao micro TK90X, da Microdigital. Quanto ao outro periférico apresentado pela empresa na última Feira de Informática, a impressora TK Printer, ainda não há previsão de chegada ao mercado.

E por falar em mercado, a chegada dos dois novos equipamentos da linha MSX não parece ter afetado as vendas dos micros da linha Sinclair, fabricados pela Microdigital. Segundo o diretor de marketing da empresa, Jorge Luiz dos Santos, os micros da Sharp e da Gradiente deram uma força ainda maior à área de microinformática,

e pelo próprio custo destes equipamentos ele se posicionaram numa faixa diferente da que é atingida pela Microdigital. "Além do que", reforçou Jorge dos Santos, "nossos micros já contam com enorme suporte a nível técnico e de software".

O diretor de marketing da Microdigital afirmou ainda que nada impede que sua empresa venha a lançar um equipamento dessa linha no próximo Natal, principalmente se se levar em conta que as empresas que lançaram micros MSX no Brasil são novas na área, não tendo tradição em informática.

## Cobra 480

A Cobra está anunciando diversos aperfeiçoamentos para o seu supermicro Cobra 480.

Estas melhorias consistem em aumentar o número máximo de linhas assíncronas, acrescentar uma unidade de fita streamer de 21 Mb para back-up e adicionar uma unidade de disco Winchester de 10,51 ou 67 Mb.

Também são anunciadas novas versões dos sistemas operacionais SOD (E.00) e MUMPS (C.02), com programas utilitários para formatação e cópia de arquivos nos novos periféricos. Estes sistemas estarão disponíveis a partir de janeiro e fevereiro.

## MSX Informática

A linha MSX conta agora com uma empresa especializada. Trata-se da MSX Informática Ltda. que atua tanto na comercialização desses equipamentos quanto na parte de suporte técnico e software para os mesmos. Inaugurada no final do ano, dentro da MSX Informática passa a funcionar este ano também o MSX Soft Club, que conta com mais de 200 programas já disponíveis para locação. A MSX Informática funciona na rua Caiubi, nº 567, tel.: (011) 872-0730, São Paulo — Capital.

## Planos para 86

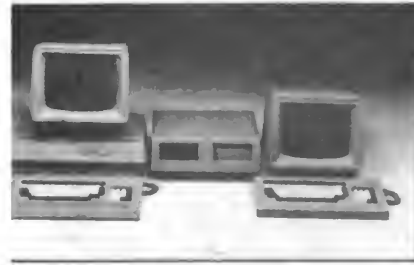
Ano novo, planos de expansão. A empresa Compumicro, que atua nos mercados do Rio de Janeiro e São Paulo, cresceu bastante ao longo de 85 e pretende diversificar suas atividades. Vários projetos têm sido estudados pela diretoria, em especial uma nova filosofia de pontos de venda e a entrada na área de treinamento.

Na área de 8 bits, — que equivale a 30% do seu volume de vendas —, a Compumicro revende basicamente UCPs da Unitron (linha Apple); drives e impressoras da Elebra; monitores de vídeo Compo e placas da Unitron e Logodata.

Mas é atuando no segmento de 16 bits, como representante do grupo Proceda, que a empresa tem realizado grandes negócios, comercializando principalmente para empresas estatais os micros Nexus 1600; Nexus 1684 (com Winchester de 10 Mb); PC 2001; XT 2002 e PC-PAQ. Também para o mercado profissional, a Compumicro dispõe da linha de comunicação (terminais e adaptadores) da Scopus.

Para todos os produtos da Proceda, além da venda existe a opção do aluguel. Este é um contrato de 30 meses, que inclui manutenção e opção de troca, o que permite ao cliente manter a atualização tecnológica de seus sistemas e reorientar seus projetos.

## Automação Industrial



Estação CAD/CAM com I-7000 PCXT

Participando da cerimônia de abertura do 2º Congresso Nacional de Automação Industrial o ministro da Ciência e Tecnologia, Renato Archer, destacou que o setor de automação deve ter o mesmo tipo de apoio que vem sendo dado à área de informática. E para isso, segundo o ministro, deve haver uma política definida que permita que os impactos da implantação da automação no mercado de mão-de-obra sejam absorvidos, e os benefícios dessa mesma implantação sejam distribuídos socialmente. Archer ressaltou

## IBAM: seminário

O IBAM vai promover nos dias 23 e 24 de janeiro o seminário "Informática: tendências tecnológicas e organizacionais". Haverá turmas das 8:30 às 12:00 e das 13:00 às 17:00 hs. Maiores informações pelo tel.: (021) 266-6622.

## Sorteio TK90X

Os vencedores da segunda rodada do concurso "Ganhe um micro TK90X", promovido pela Microdigital em conjunto com MICRO SISTEMAS, são Marcelo Roberto Gimenez, do Rio de Janeiro, e Carlos Mello Mesquita, de Porto Alegre.

## Conexão Bolsa

A Clappy, empresa carioca especializada na venda de equipamentos, software e serviços está oferecendo um novo pacote de software: Conexão Bolsa.

Desenvolvido pela B & C Consultores, Conexão Bolsa permite a qualquer usuário dos micros Apple, TRS-80 e IBM-PC, ligar o seu equipamento aos computadores da BVRJ, possibilitando obter informações sobre o mercado de ações.

A Clappy fica na Av. Rio Branco, 12 loja. Tel: (021) 253-3395.

## Cartão de referência

Foi lançado recentemente pela Ponto Editorial o cartão de referência para o micro Apple II e similares. O cartão possui 16 páginas e substitui o manual do proprietário, de 320 páginas, e aborda os principais recursos do microcomputador tais como: Integer Basic, Applesoft, DOS 3.3 e Assembly 6502. O cartão de referência pode ser encontrado nas principais livrarias e custa Cr\$ 40 mil, e pode ser pedido também diretamente à Ponto Editorial, rua Caetés, 252 — Perdizes — CEP: 05016, tel.: (011) 864-3499 São Paulo.

## Cursos BASIC

Em fevereiro, a Datamicro vai promover cursos de BASIC Básico e BASIC Avançado. O Básico começa dia 4, com turmas às segundas, quartas e sextas, em diversos horários, e o curso de BASIC Avançado terá início dia 5, com dias de aula a combinar. Informações pelo tel.: (021) 511-0395.

## Arquitron

O arquiteto José Eduardo Maluf de Carvalho, colaborador de MICRO SISTEMAS e um dos primeiros a introduzir o uso do micro em sua área, está cada vez mais ligado à Informática. Prova disso é que acaba de abrir uma loja para comercialização de computadores. Mas segundo José Eduardo, na Arquitron, mais do que para o comércio de micros, a ênfase é dada a parte de serviços, tais como desenvolvimento do software para as mais variadas áreas e a criação de um clube de usuários do TK90X. A Arquitron fica na Av. Faria Lima, 1.684 sala 23, Shopping Vitrine Iguatemi, São Paulo.

## Atendimento Prológica

Os usuários dos computadores pessoais da Prológica contam agora com um serviço de reparos especial. Trata-se do Atendimento Rápido Prológica, que garante o conserto quase que imediato da máquina, com prazo médio de apenas 30 minutos de espera e garantia total por 45 dias. Além disso os preços do novo serviço são fixos: CP-200, 1,8 ORTN; CP-300 e 400, 2,8 ORTN e 4 ORTN para o CP-500 modelo 021. Para meios deste ano o Atendimento Rápido Prológica deverá ser estendido à área de Sistemas e impressoras. O ARP funciona na rua Fidêncio Ramos, 302, no bairro de Vila Olímpia — São Paulo, Capital.

## Novas instalações

A Maquis transferiu parte de suas dependências para a Rua Inhambu, 1271, tel.: (011) 543-3221, em Moema, São Paulo. No antigo prédio da Av. Imarés ficou apenas a fábrica.

No Rio, a Maquis também mudou de endereço. A empresa instalou seu novo show-room na Av. Calógeras, nº 6B, sobrelojas, tel.: (021) 240-4934, Centro.

## Padrões brasileiros

Com o objetivo de discutir a criação de padrões brasileiros para o setor de Informática, difundir os resultados alcançados até o presente, discutir a integração desse esforço à importância da Certificação de Qualidade Industrial e colher subsídios para os trabalhos futuros, a ABNT — Associação Brasileira de Normas Técnicas —, através do seu CB-21, realizou de 18 a 20 de novembro no Hotel Glória (Rio de Janeiro), o Seminário de Normalização Técnica e Qualidade Industrial em Informática.

Constituído de vários painéis e cursos de curta duração, o Seminário destacou a importância do setor de Informática — que este ano faturará mais de US\$ 2 bilhões — para o país, enfatizou a necessidade de padronizar alguns recursos mínimos dos sistemas operacionais (chamadas de supervisão, bibliotecas, arquivos, etc.) e alguns aplicativos de uso geral, visando beneficiar a indústria, o comércio e principalmente o usuário. Os participantes também afirmaram a necessidade de que o Brasil procure soluções próprias para os seus problemas e não apenas copie programas e sistemas importados.

## Outras empresas na trilha do MSX

Duas empresas do mercado de Informática estudam seriamente a possibilidade de entrar na fabricação de micros da linha MSX. São elas a Dynacom e a Racimec. A Dynacom avalia com calma a hipótese de mais um produto, visto que a empresa teve problemas com as placas do micro MX-1600 — compatível com a linha Color, da Radio Shack, e com um nome bastante sugestivo, para quem pretende investir no padrão japonês. O Color da Dynacom estava previsto para chegar às lojas no mês passado, aproveitando as vendas de fim de ano, mas devido aos tais problemas de fabricação, teve seu lançamento adiado para janeiro, segundo a diretoria de Marketing. Quanto ao MSX, poderemos ter uma surpresa, quem sabe numa dessas feiras de entre-safrá.

Já na Racimec — que não brinca em serviço e apóia-se num confortável parque instalado de seus sistemas profissionais e terminais lotéricos — as pesquisas estão bem adiantadas. Consta que já existem equipes de desenvolvimento e software debruçadas sobre diversos modelos do MSX, somente à espera do sinal verde para acionar os motores de fabricação. E o sinal poderá vir em breve, pois segundo o presi-

dente Simão Brayer, tudo depende de serem resolvidos os impasses de fabricação na Zona Franca de Manaus, para que haja uma "competição limpa". Recentemente, a Gradiente teve problemas para o lançamento do micro Expert, visto que uma auditoria por parte da SEI constatou que havia planos de fabricação do micro, contando com os incentivos da Suframa já garantidos a um projeto de videogame aprovado pelo Governo e que, afinal, nem vai ser fabricado. Detestado o "engano", a solução parece ser a mesma da Epcom, que já fabrica em Manaus o Hot Bit, porém sem contar com as regalias daquela região.

O Presidente da Racimec, contudo, parece não estar convicto desta ser a melhor solução, pois, disse ele, "pelo que conheço da burocracia e dos processos de controle, esta situação pode gerar indefinições e até facilidades para quem já está lá". De qualquer forma, embora o impasse maior esteja na certeza de se conseguir preço competitivo, Brayer não esconde a empolgação com o produto — "Não há dúvida de que o MSX é um micro profissional" —, e como a Racimec é movida em grande parte, por seu entusiasmo, dá pra apostar.



Mesmo sendo versátil, o sistema operacional do TRS Color não possui recursos para combinar textos com gráficos em alta resolução. Porém, se você utilizar as rotinas descritas neste artigo isto será fácil.

# Caracteres no Color

Cláudio Costa

Uma das maiores limitações dos micros compatíveis com o TRS Color é, sem dúvida, a impossibilidade de se combinar texto com alta resolução. Nesses equipamentos, o hardware para geração de vídeo seleciona alternativamente o modo de apresentação entre texto ou gráfico: no primeiro modo, um byte de vídeo é traduzido como um caráter; no segundo, esse mesmo byte é interpretado como um pixel ou ponto da tela gráfica (veja o artigo "Vídeo no Color", MS nº 48).

É possível concluir que tal limitação, até certo ponto, se impõe em razão do sistema operacional não ter sido dotado, a princípio, de rotinas capazes de traduzir as informações das telas gráficas na forma de caracteres. Logo, se quisermos aliar texto à alta resolução, o que temos a fazer é simplesmente escrever essas rotinas. Há várias maneiras de se fazer isso, tanto em BASIC como em linguagem de máquina.

Um método clássico é o que utiliza o comando **DRAW**. Considere uma matriz da letra A, como a ilustrada na figura 1.

Portanto, se você desejasse imprimir o caráter da figura 1, a instrução **DRAW** viria acompanhada de: U5R4D2NL4D3, o que pode ser visto na figura 2. Este comando significa: suba cinco pontos, ande quatro para a direita, dois para baixo, quatro para a esquerda, sem atualizar a posição de impressão e, finalmente, mais três pontos para baixo (o funcionamento deste comando está claramente explicado no manual). Digite o programa abaixo e confira o resultado

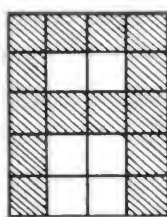


Figura 1

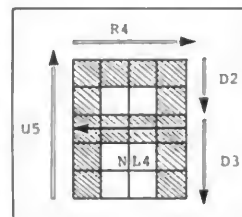


Figura 2

na tela de alta resolução:

```
10 PMODE4:COLOR5,5:PCLS:SCREEN1,
1
20 DRAW"U5R4D2NL4D3"
30 GOTO30
```

Como o argumento de **DRAW** pode ser uma string, é possível estender este processo para a impressão de mensagens inteiras; basta definir as instruções **DRAW** para os caracteres e acessá-las através de uma matriz. Este método implica em estabelecer parâmetros para todo o alfabeto, trazendo porém, em contrapartida, uma grande flexibilidade. Com efeito, não precisamos nos restringir ao conjunto de caracteres convencionais do micro; podemos criar letras minúsculas, acentuação da língua portuguesa, caracteres especiais etc., tudo no formato que desejarmos.

Certos cuidados são necessários, se a intenção é criar um sistema que funcione em todos os PMODEs. Experimente, no exemplo anterior, trocar o PMODE 4 da linha 10 por PMODE 0. Rode novamente o programa, você notará que a perda de definição do desenho se deve ao fato do ponto, na baixa resolução, ser mais alto do que largo. Para funcionar em todos os PMODEs, um sistema

deve, portanto, prever as diferenças existentes entre os diversos modos gráficos.

A sub-rotina apresentada na listagem 1 imprime caracteres maiúsculos em qualquer PMODE, bastando para tal comandar **GOSUB 5**. A mensagem (máximo de 25 dígitos) deve estar contida na variável **TE\$**; **PX** define a coluna e **PY** a linha de início da impressão. Apenas o intervalo de 0 a Z foi utilizado (**CHR\$ 48** a **90**, mais um espaço), logo, é só subtrair 47 do código ASCII de cada um dos caracteres a serem impressos para se obter a posição do desenho correspondente na matriz (linha 6). Antes, porém, são necessários alguns cálculos com o objetivo de compatibilizar a posição relativa do traço nos diversos PMODEs. Assim, **PY** deve ter um valor par, enquanto **PX** deve ser ímpar; as comparações lógicas da linha 5 verificam a consistência desses valores. O espaçamento entre os caracteres (**SP**) é calculado como sendo aproximadamente o dobro do valor da escala empregada, o qual pode ser referenciado por meio de **PEEK (233)** (linha 5). Algumas letras, como **M** e **W**, no entanto, requerem mais espaço para serem representadas; este espaço é indicado pelo parâmetro **B**, no final da instrução **DRAW** (linha 7). Note que alguns caracteres foram redefinidos: o sinal **<**, por exemplo, imprime um Ç. Veja na figura 3 um quadro com os caracteres redefinidos na rotina. Qualquer caráter fora da faixa válida será interpretado como um espaço.

Para ver a rotina funcionando, rode o programa abaixo, pois ele imprime a

mensagem em letras ciano sobre fundo cinza (**COLOR 6, 5**) em média resolução (**PMODE 3**):

```
10 PMODE3:COLOR6,5:PCLS:SCREEN1,
1
20 TE$="CARACTERES":PX=73:PY=90:
GOSUB5
30 GOTO30
```

A escala em que serão desenhados os caracteres pode assumir qualquer valor numa faixa de 4, 8, 12, 16 etc. Para ajustá-la, deve ser usado normalmente o parâmetro **S** do comando **DRAW**, tal como mostra o exemplo a seguir:

```
10 PMODE1:COLOR5,8:PCLS:SCREEN1,
1
20 TE$="CARACTERES":PX=43:PY=96:
DRAW"SB":GOSUB5
30 GOTO30
```

Note, por fim, que você pode exibir numa mesma tela caracteres de diferentes formatos. Basta passar os caracteres de um PMODE para outro:

```
10 TE$="CARACTERES"
20 PMODE1:COLOR8,5:PCLS:SCREEN1,
1:PX=41:PY=76:DRAW"SB":GOSUB5
30 PMODE3:COLOR6,5:PX=79:PY=56:DRAW"SA":GOSUB5
40 FORT=1:GOTO30:NEXT
50 PMODE4:SCREEN1,1:PX=81:PY=90:
GOSUB5
60 GOTO60
```

Com base no que foi visto, você poderá adaptar a rotina de acordo com suas necessidades. Se você quiser usá-la como está, digite o comando **CLOAD** seguido da rotina e inicie a numeração de seu programa a partir de 10. Não es-

queça, nesse caso, de ajustar o parâmetro **CLEAR** da linha 1 de acordo com o programa principal.

## MODIFICANDO O FORMATO DA TELA

Uma outra característica que pode passar despercebida na primeira aproximação, mas que é capaz de se tornar objeto de alguns aborrecimentos no convívio com estes micros é, certamente, a pequena capacidade de sua tela de texto, capaz de acomodar apenas 32 colunas por 16 linhas. Tal divisão, contudo, não é gratuita: como a memória do Color é dividida em páginas de 1536 bytes cada, e os 1024 bytes iniciais da primeira página estão reservados para as variáveis do sistema, sobram apenas 512 bytes para alocar a tela de texto. Como se vê, o arranjo de 32 colunas por 16 linhas é o que melhor se adapta a esta organização de memória (note que 32 x 16 perfazem exatos 512 bytes).

CARÁTER DO MICRO	CARÁTER DA ROTINA
?	,
!	Ç
>	À
?	.
@	.

Figura 3

Uma solução para este problema — e que resolveria ainda a incompatibilidade de de texto com alta resolução — seria uma rotina que habilitasse a tela de alta resolução gráfica a funcionar também como tela de texto, mantendo-se o princípio do emprego de caracteres definidos pelo programador. Isto é possível graças a maneira como foi projetado o sistema operacional desses equipamentos, e embora tal rotina precise ser escrita em linguagem de máquina, o conceito que envolve sua operacionalidade é bastante simples.

Na verdade, a rotina do sistema operacional responsável pela impressão de um caráter, localizada em \$A282, executa inicialmente um **JSR** (Jump do Sub Routine, que equivale a um **GOSUB** do BASIC) para uma posição da RAM — o endereço 359 decimal, mais precisamente. Neste momento, o caráter a ser impresso está contido no registrador A. Esta posição contém apenas uma instrução de desvio (**JUMP**) para outro endereço da ROM, onde continua o processamento; logo, para interceptar a rotina de impressão, tudo o que temos a fazer é colocar um **JUMP** para a nossa rotina na posição apropriada da RAM, salvar os registradores naquele ponto, executar a rotina de impressão do caráter na tela gráfica, recuperar o conteúdo dos registradores e comandar um **JUMP** para

## COMUNICAÇÃO ENTRE MICROS PARA TRANSFERÊNCIA DE ARQUIVOS

O **TRANSFERE** é a solução para o seu problema de ter vários micros com formatos de disquete ou sistemas operacionais incompatíveis.

Com o **TRANSFERE** você pode transferir arquivos entre dois micros através de uma ligação por cabo. O **TRANSFERE** usa as interfaces seriais dos seus micros, podendo funcionar com velocidade de transmissão de até 9600 bauds.

Na tabela ao lado você encontra as características mais importantes do **TRANSFERE**. Use-a para facilitar a sua decisão de compra, comparando o **TRANSFERE** com os concorrentes.

Computadores já suportados: **POLYMAX**, **EDISA**, **DISMAC**, **TELSIST**, **ITAUTEC**, **PROLOGICA**, **MICROTEC**, **SCOPUS**, **UNITRON**, **MICROENGENHO**, todos os compatíveis com o **APPLE** usando placa **CP/M** e todos os compatíveis com o **IBM PC** (consulte-nos se o seu computador não está listado, novas versões estão sendo feitas).

**PREÇO:** 20 ORTNs para compatíveis com o **APPLE II**,  
30 ORTNs para micros **CP/M**  
40 ORTNs para micros **MS-DOS**  
(compatíveis com **IBM PC**)

### Compare você mesmo o TRANSFERE com os concorrentes

CARACTERÍSTICAS	TRANSFERE	
Permite a transferência de arquivos entre micros CP/M e compatíveis com IBM PC	SIM	
Faz a transmissão usando um protocolo de verificação e correção automática de erros	SIM	
Transmite qualquer tipo de arquivo, como fontes de programas, textos, arquivos de dados, programas executáveis, etc.	SIM	
Permite transmitir vários arquivos com um único comando, usando a convenção de ? e * para especificar o grupo.	SIM	
A transmissão pode ser feita através de ligação direta por cabo ou através de linhas telefônicas usando modems.	SIM	
Para transferência de arquivos basta dar o comando em um dos micros ligados, o outro responde automaticamente.	SIM	
Permite acessar qualquer drive em qualquer um dos micros, podendo listar diretórios do micro local ou do remoto.	SIM	
Permite emular um terminal com protocolo TTY para ligação com CIRANDÃO, DISQUE-BOLSA, ARUANDA e outros serviços.	SIM	
Acompanha o programa um manual completo, que explica detalhadamente a sua utilização	SIM	

**INTELISOFT**

INTELISOFT Projeto e Desenvolvimento de Sistemas S/C Ltda.  
Praia do Flamengo 66 Sala 1104 CEP 22210 Tel. (021)265 3346 - Rio de Janeiro - RJ



## Listagem 1

```

1 CLEAR200: DRAW "S4" = DIMC$(43): FO
RJ=1 TO 43: READ C$(J): NEXT J TO 10
2 DATA C$(20): NG3D5, BR3U7G2, BU6E
R3FD2GL3GD2R5, BU6ER3FDGNLFD2GL3H
, BR3U7G4R6, BU7NR5D3R4FD2GL3H, BU3
R4FDGL3HU3E3R, E6UL6, XC$(4): U2EHU
, BRRE3L4HU2ER3D2, UBU4U, UD2, XC$(
20): BD6BL2D2, BU2R4BU3L4, XC$(18):
, BU9BLL3, BU3R4, U, U6ER3FD3NL5D3, XC
$(33): FD2GL4
3 DATA BR4NEL3HUSER3F, U7R4FD5GL4,
XC$(23): D4R5, U7R5BD2BGL4, XC$(20)
: BD3NL2D2, U7D3R5U3D7, BR2U7NL2NR
2D7R2, BRNHR3EU6, U7D3R2NE3F3D, NU7
R5, U7F3RE3D7B, U7F6DNU7, XC$(20): D
5, U7R4FDGNL4, XC$(32): BL2F2, XC$(3
3): L2F4, R4EU2HL3HUER4, BR3U7L3R7B
, BU7D6FR3EU6, BU7D4F3R3U4B
4 DATA NU7E3R3F3U7B, UE5UBL5DF5D, BR
3U4NH3E3B, BU7R6G7R7
5 PX=PX OR 1: PY=PY-(PY AND 1): SP=P
EEK(233)*2+2
6 FOR J=1 TO LEN(TES): CH=ASC(MID$(T
ES, J, 1))-47: IF CH(00RCH) 43 THEN CH=
0
7 DRAW "BM" + STR$(PX) + ", " + STR$(PY)
+ CH$(CH): PX=PX+SP-2*(RIGHT$(CH$(CH
), 1) = "B"): NEXT J: RETURN
8 'VARIÁVEIS USADAS: C$, CH, J, PX,
PY, SP, TES
10 '*****

```

## Listagem 2

```

10 '*****
15 ' * TEXT+ BIN
20 '*****
25 ' * Combina texto com alta
30 ' * resolucao e altera o
35 ' * formato da tela para
40 ' * 32 colunas X 24 linhas
45 '*****
50 ' * Claudio Costa * MS *
55 ' * 09/1985 *
60 '*****
65 '
70 CLEAR200, 256*PEEK(116)-768: :=
256*PEEK(116)-768: F I+1016
75 CLS: (KEYFNQ, 1): FORB I TO 1: REA
DA$=A: AI (-XH+A$): C=1: A=POKEB, A
: N: XT
80 IF (1) 91655 THEN PRINT@203, "
ERRO *": POKE137, 0: CHD
85 PRINT@101, "PREPARE O GRAVADOR
L TECLA ENTER": IF INKEY$ = "T": JB
5
90 CSAVEN "TEXT", I, F, I
95 '
100 DATA D1, A2, EF, BD, DD, F6, 13, BD
, DD, A9, DF, A9, FE, D1, 6B, EF, BD, D1, 0
, 7, C3, DD, DD, EF, 1F, D1, 6B, 33, BD, D1,
5F, DF, FC, C1, 12, 64, DF, FA, C6, D4, BD
, 96, 2C, C6, C1, 07, H3, DF, B2, BD, 76, B
D, 95, CF, 33, BD, D1, 21, A6, CD, 27, 07,
43, AD, 9F, AD, D2, D1, F5, F1, D1, 6B, EF

```

```

, BD, DD, 5B, 30, BD, DD, 26, BF, D1
105 '
110 DATA 6B, CE, DD, C8, 34, 40, 7E, AF
, 54, AB, BA, A7, AB, D4, F2, B2, 96, 9C, B
D, 90, DF, AC, 96, BC, BD, 9A, 92, 9E, BC,
DF, CE, C6, C7, C9, F2, DD, 34, 56, B1, DC
, 26, 05, BD, 2H, 17, DD, DD, 96, 77, 27, 1
1, 9E, FA, BD, A7, D3, 8E, D1, 52, B6, FF,
A7, 80, 8C, D1, 5A, 26, F9, 96, 76, 27, DA
, CC, 0B, 04, 97, 8C, D7, BE, BD, A9
115 '
120 DATA 56, 35, 56, 7E, BC, F1, 34, 16
, BD, 95, 42, 9E, BA, 9F, FE, 35, 96, 7D, F
F, 22, 2A, 43, B1, 9E, 27, 10, B1, 3D, 26,
0B, 34, 56, 4F, 17, DD, 9B, 35, 56, B1, 97
, 26, 02, BD, DA, B1, 40, 26, 29, BC, AB, F
5, 26, 24, 34, 16, BD, C3, E4, 10, B3, 02,
FF, 10, 22, 37, 5F, 1F, D1, BC, DD, 20, 25
, 0B, 30, 88, E0, C3, DD, E0, 20, F3
125 '
130 DATA D3, BA, DD, FE, 35, 16, B4, 20
, 7E, AA, 1A, DD, 6F, 26, 17, 34, 56, B1, 0
B, 26, 14, 4F, BD, 56, DC, FE, 5A, 2A, D3,
B3, D1, E0, DD, FE, BD, 46, 35, D6, 7F, B2
, 73, B1, E0, 26, 09, 4F, BD, 3E, DC, FE, 4
C, 1F, 20, 14, B1, 20, 25, EA, B1, 7A, 22,
E6, BD, 2C, DC, FE, 5C, C1, 20, 25, D3, C3
, DD, E0, DD, FE, 10, 93, 07, 25, 11
135 '
140 DATA DA, FE, 9E, BA, EC, 09, 01, DD
, E0, B1, 9C, FE, 26, F4, 9F, BA, BD, 95, A
E, 96, 1C, 77, DA, 20, BD, B6, 61, BD, 0, 0,
C6, 07, 3D, D3, FC, 1F, D1, DE, FE, B6, 07
, E6, BD, DD, B3, 26, D1, 53, E7, C4, 33, C
B, 20, 4A, 26, F1, 39, FF, FF, FF, FF, FF,
FF, FF, F7, F7, F7, F7, F7, FF, F7, E0, EB
, EB, FF, FF, FF, FF, EB, EB, C9, F1
145 '
150 DATA C9, EB, E0, F7, E1, DF, E3, FD
, C3, F7, C0, C0, FB, F7, EF, D9, F7, D
7, D7, FF, 05, DB, E5, 17, F7, EF, FF, FF,
FF, FF, F7, EF, DF, DF, DF, FF, F7, F7, FB
, FD, FD, FB, F7, FF, F7, E3, 21, E3, F
7, FF, F7, F7, F7, F7, FF, FF, FF, FF,
FF, 17, E7, F7, EF, FF, FF, F7, C1, FF, FF
, 1, F1, FF, 11, FF, 11, C7, C7, FD
155 '
160 DA A FD, FB, F7, FF, DF, DF, E7, DB
, DB, DB, DB, E7, F7, F7, F7, F7, F7,
F7, E3, E3, DD, FD, 13, DF, DF, C1, E3, DD,
FD, FB, FD, DD, F1, FB, F3, FF, C1, FB, FB
, FB, E1, DF, C3, FD, FD, DD, E3, E3, DF, D
F, C3, DD, DD, F3, C1, FD, FB, F7, EF, DF,
DF, 13, DD, DD, E3, DD, DD, F3, C3, DD, DD
, E1, FD, FD, 13, FF, E7, E7, F1
165 '
170 DATA E7, FF, E7, E7, F1, E7, C7, F7
, E7, FB, F7, EF, DF, DF, F7, FB, FF, FF, C
1, FF, C1, F7, FF, EF, F7, F7, FD, FB, F7,
FF, E7, DB, FB, F7, F7, F1, F7, E7, DD, FD
, ED, D5, D5, E3, F7, FD, DD, C1, DD, D
D, C3, CD, ED, E3, ED, ED, C3, E3, DD, D1
, DF, DF, DD, E3, C3, LD, ED, ED, ED, E3,
C1, DF, DF, C3, DF, DF, C1, C1, DF
175 '
180 DATA DF, C3, DF, DF, DF, E1, DF, DF
, D9, DD, DD, E1, DD, DD, DD, C1, DD, DD, D
F, C3, F7, F7, F7, F7, F7, F7, F7, F7, FD,
FD, DD, DD, E3, DD, DB, D7, CF, D7, DB, DD
, DF, DF, DF, DF, DF, DF, C1, DD, C9, D5, D
5, DD, DD, DD, DD, DD, DD, D9, DD, DD, D
C1, DD, DD, DD, DD, DD, C1, C3, DD, DD, C3
, DF, DF, DF, F3, DD, DD, DD, D5, DB
185 '
190 DATA E5, C3, DD, DD, C3, D7, DB, DD
, E3, DD, EF, F7, FB, DD, E3, C1, F7, F7, F
7, F7, F7, DD, DD, DD, DD, DD, DD, E3,
DD, DD, DD, E8, E8, F7, F7, DD, DD, DD, D
5, C9, DD, DD, DD, DD, E8, F7, E8, DD, DD, D
D, DD, E8, F7, F7, F7, C1, FD, FB, F7,
EF, DF, C1, C7, DF, DF, DF, DF, DF, C7, E7
, DB, DB, E7, FF, E7, FF, E3, FB, FB
195 '
200 DATA FB, FB, FB, E3, F7, E3, D5, F7
, F7, F7, FF, F7, FB, B1, FB, F7, FF, F
F, FF, FF, FF, FF, FF, C1, FF, FF, C7, FB,

```

```

C3, BB, C7, DF, DF, D3, DD, DD, DD, C3, FF
, FF, E1, DD, DF, DD, E3, FD, FD, E1, DD, D
D, DD, E1, FF, FF, E3, DD, C1, DF, E1, F3,
, 11, L, EF, EF, EF, FF, FD, E3, DD, E3
, FD, E3, DF, DF, D3, DD, DD, DD, DD
205 '
210 DATA F7, FF, L7, F7, F7, F7, E3, FB
, FF, FB, FD, FB, FD, E7, DF, DF, DB, D7, U
7, DB, DD, E7, F7, F7, F7, F7, F7, E3, FF,
FF, B9, B6, B6, B6, DE, FF, FF, D3, DD, DD
, DD, DD, FF, FF, E3, DD, DD, DD, E3, FF, D
3, CD, DD, C3, DF, DF, C3, CD, DF, DF, FF, FF
, FD, FD, FF, FF, D3, CD, DF, DF, FF, FF,
C3, DF, C3, FD, 13, EF, EF, B3, 11
215 '
220 DATA E1, ED, F3, FF, FF, DD, DD, DD
, DD, 11, F1, FF, DD, DD, DD, EB, F7, FF, F
F, BL, B6, B6, B6, C7, 11, FF, C3, EB, F7,
EB, DD, FF, FF, DD, DD, E1, FD, E3, FF, FF
, C1, CD, 13, DF, C1

```

o endereço originalmente apontado por aquela posição da RAM. Evidentemente, o programa não se resume apenas nisso; uma série de outras funções precisam ser implementadas, tais como: impressão do cursor, delete, backspace, execução do SCROLL automático quando a tela estiver cheia etc.

É bom saber, entretanto, que é possível escrever um programa que execute todas essas funções usando menos de 300 bytes. O maior consumo de memória, na realidade, está relacionado com a tabela de que tal programa necessita para desenhar os caracteres. Não é possível, a exemplo do que fazem os usuários do ZX81, ler a tabela de definição de caracteres do micro, porque essa tabela não está localizada na ROM do sistema operacional e sim na ROM do microprocessador de vídeo, o VDQ.

Como a definição do formato dos caracteres fica a critério do programador, não existem, a rigor, restrições quanto ao número de linhas ou colunas que podem ser utilizadas pelo novo sistema. Alguns critérios, todavia, devem ser seguidos e um deles diz respeito à legibilidade dos caracteres. Isto pode parecer óbvio, mas muitos programas comerciais desenvolvidos com o objetivo de ampliar a capacidade da tela de texto do Color Computer pecam nesse aspecto. Enquanto tecnicamente não há problema em operar com 64 caracteres por linha, por exemplo, na prática, o diâmetro dos caracteres não passaria de três ou quatro pixels, considerando-se que o limite horizontal da tela é de 256 pontos (256/64 = 4). Os caracteres normais do micro têm uma largura de oito pixels, por onde se percebe que a imagem, num televisor comum, seria no mínimo sofrível.

O programa apresentado na listagem 2, cria uma tela de texto de 32 colunas por 24 linhas, ou 768 posições. Este formato é mais ou menos padrão para equi-

pamentos deste porte, e foi adotado em função da maioria das dificuldades de manipulação da tela de texto estarem relacionadas com o número de linhas disponíveis. Este formato permite ainda que seja mantido o desenho original dos caracteres do micro, já que apenas o espaçamento entre as linhas é modificado. O programa imprime letras maiúsculas e minúsculas em vídeo normal ou inverso, além de implementar opcionalmente novas funções, como repetição automática de teclas e feedback auditivo (beep).

## COMO USAR O PROGRAMA

Digite cuidadosamente a listagem 2 e grave-a em fita. Em seguida, dê RUN; não havendo erro, a tela permanecerá cor de laranja durante, aproximadamente, 15 segundos. Prepare o gravador, dê ENTER e grave a versão final do programa. Para carregar e executar, digite a seguinte sequência em forma de comando direto:

CLEAR 2000:CLOADM:EXEC

O próprio programa se encarrega de reservar espaço no topo da RAM. Apenas 262 bytes, mais 637 bytes de tabela são usados; as rotinas de inicialização

são apagadas a fim de economizar memória. Os comandos CLS e PRINT e a tecla CLEAR funcionam normalmente, porém, a rotina não imprime caracteres gráficos. Para manter a compatibilidade com programas que utilizam esse recurso, é possível reverter à tela de texto normal digitando-se SCREEN 0. É possível ainda fixar a tela de texto original na cor de laranja; experimente comandar SCREEN 0,1. Para retornar à alta resolução, digite SCREEN 1.

A instrução COLOR, além de definir a cor que será usada por CLS e pela rotina de SCROLL, estabelece também a cor em que serão impressos os caracteres. Dessa forma, SCREEN 1,0: COLOR 0,1: CLS imprime caracteres em preto sobre fundo verde, enquanto SCREEN 1,0: COLOR 1,0: CLS imprime caracteres em verde sobre fundo preto (as instruções SCREEN e CLS, obviamente, não são obrigatórias). Já as rotinas de beep e auto-repeat podem ser ativadas da seguinte forma:

POKE 118,1 — ativa beep  
POKE 118,0 — desativa beep  
POKE 119,1 — ativa repeat  
POKE 119,0 — desativa repeat

O valor armazenado nos endereços 250 e 251 é assumido como o intervalo

de tempo que uma tecla aguarda ao ser pressionada, antes de ser repetida. Este delay é ajustado inicialmente em 6500, mas pode ser modificado pokeando-se qualquer outro valor nestes endereços.

Da mesma maneira que SCREEN, é possível modificar também o PMODE em uso. Note, contudo, que não é seguro trocar o PMODE quando a impressão estiver situada na última linha da tela, pois a rotina de SCROLL perde os valores de referência e pode ocorrer perda de controle do cursor. Quanto à instrução RESET, ela não elimina o programa da memória mas faz o computador retornar à tela de texto normal. Para voltar à tela de alta resolução, digite SCREEN 1. Um RESET geral do sistema é obtido por POKE 113,0: EXEC 40999.

Embora seja curto, este programa executa várias operações, e uma análise dos procedimentos de programação envolvidos fica para outras edições de MICRO SISTEMAS. Por ora, vale constatar que com criatividade e um pouco de técnica, não há limitação que não possa ser superada, às vezes até com vantagens.

Cláudio Costa é desenhista free-lancer e tem como hobby a programação de microcomputadores.

## CONTABILIDADE ON-LINE PARA MICROCOMPUTADORES

A Contabilidade INTELSOFT é a única que realmente funciona em modo on-line. A qualquer momento você pode fazer consultas no vídeo para obter saldos de qualquer conta e lançamentos em formato de razão.

A Contabilidade INTELSOFT funciona tanto em micros de 8 bits, com sistema operacional CP/M, como em micros de 16 bits, compatíveis com o IBM PC. Todas as versões são funcionalmente idênticas podendo os mesmos arquivos serem usados por qualquer versão. Isso significa que ao mudar de computador você continua trabalhando da mesma maneira e aproveita integralmente o trabalho já feito.

Na tabela ao lado você encontra as características mais importantes da Contabilidade INTELSOFT. Use-a para facilitar a sua decisão de compra, comparando-a com as concorrentes.

PREÇO: 150 ORTNs para micros CP/M  
200 ORTNs para micros MS-DOS  
(compatíveis com IBM PC)

\*Caso você queira experimentar o sistema antes de comprá-lo, a INTELSOFT fornece um pacote de demonstração por apenas 15 ORTNs. O pacote de demonstração é uma versão completa do sistema, sem nenhuma limitação, acompanhada do manual. O valor do pacote de demonstração pode ser aplicado integralmente, em ORTNs, na compra do sistema, em um prazo máximo de 60 dias.

## compare você mesmo a nossa contabilidade com os concorrentes

CARACTERÍSTICAS	INTELSOFT	
Totalmente controlado por menus e realmente on-line.	SIM	
Número máximo de contas	65535	
Número máximo de lançamentos	65535	
Máximo de algarismos por valor	15	
Máximo de graus no plano	8	
Código de conta flexível e definido pelo usuário com número variável de dígitos por grau.	SIM	
Lançamentos de partida simples ou dobrada	SIM	
Históricos livres e padronizados	SIM	
Permite trabalhar com ou sem centavos	SIM	
Permite consolidar os dados de diversas empresas ou filiais	SIM	
Emite relatórios especiais para corretoras e distribuidoras de acordo com os modelos do Banco Central	SIM	
Permite que os dados do sistema sejam convertidos para serem usados por pacotes como o dBASE II ou III, LOTUS 1-2-3, VISICALC ou por programas escritos em qualquer linguagem.	SIM	



Você possui um TK90X e quer desenvolver o seu próprio adventure?  
Se a resposta for sim, este editor poderá auxiliá-lo nesta missão.

# Editor de Adventures

Paolo Fabrizio Pugno

**A**ntes de mais nada, vamos botar os pingos nos devidos iis: os programas "Editor" e "Adventure" foram elaborados e adaptados para a linha ZX Spectrum a partir de uma idéia semelhante publicada na revista italiana "Microcomputer", para a linha Apple.

O assunto é adventure, este fascinante tipo de jogo que põe a dura prova nosso raciocínio, estratégia, paciência, e, às vezes, até mesmo sorte. Jogá-lo é uma coisa, elaborá-lo é outra totalmente diferente, onde devemos levar em consideração todas as possíveis situações que poderão ser enfrentadas pelo programa. Em resumo, não é nada fácil, embora não seja impossível. E tem mais, para cada adventure que escrevermos, teremos que bolar um programa gerenciador específico, que manipule os dados necessários. Isto faz com que horas e mais horas sejam perdidas na depuração de um maldito programa. Às vezes, só de pensar em enfrentar tais dificuldades, desistimos logo de saída. Mas, que tal um programa universal, isto é, que sirva para jogar todos os adventures que escrevermos? Tudo o que temos a fazer é "ensinar-lhe" as regras do jogo. Porém, é necessário que os dados lhe sejam fornecidos de uma maneira padronizada, e é aqui que entra em cena o "Editor".

Na realidade, são dois programas: com o "Editor" ensinamos as regras do jogo à máquina, que vai organizá-las de

uma forma que o "Adventure" possa manipulá-las. Os dados são salvos em fita, através da gravação de matrizes numéricas e alfanuméricas, que serão posteriormente carregadas pelo "Adventure". Desta forma, podemos inventar o adventure que quisermos, sem nos preocuparmos com o programa gerenciador. Como jogar um adventure feito por você mesmo perde logo a graça, a sugestão é realizar trocas entre amigos que possuam um ZX Spectrum ou TK90X; essa troca consistiria nas matrizes salvas em fita pelo "Editor".

## EDITOR

Antes de rodar o "Editor", é obrigatório anotar tudo o que for necessário; fazer um mapa; dar números aos locais (começando por 1) e aos objetos; anotar os verbos e seus complementos; enfim, tudo o que depois será pedido pelo programa.

Dado o RUN, o "Editor" irá pedir o nome do jogo, o qual não deverá ter mais de dez caracteres, pois será utilizado como nome de uma matriz na hora de gravar os dados em fita. O programa então perguntará se o adventure é por pontos. Teclando S será pedido o escore a ser atingido para que o jogador alcance a vitória. Logo a seguir, devemos introduzir o número total de locais de jogo, após o que será pedida a descrição

de cada local. Essa descrição não deve ultrapassar três linhas (96 caracteres); o que, no entanto, pode ser alterado na linha 60, mas cuidado para não desperdiçar memória. Após introduzirmos cada descrição, devemos indicar que local é atingido quando nos movimentamos nas seis direções possíveis (Norte, Sul, Leste, Oeste, Suba e Desça). Introduzindo o zero para uma direção qualquer, iremos provocar o aparecimento da mensagem "não posso ir nessa direção", quando a mesma for executada nesse local; introduzindo o número do próprio local, não haverá deslocamento. Note-se que não é obrigatório nos locomovermos para locais vizinhos (no mapa). Aliás, quanto mais complicado for o mapa, mais interessante o adventure se tornará.

Na segunda parte do programa, é pedido o número de objetos existentes no jogo e, depois, o nome de cada um (sem utilizar artigo) e sua localização inicial. Aqui, há um truque: o local zero não existe no jogo, e serve para colocarmos temporariamente objetos que devem desaparecer ou para guardarmos outros que irão surgir posteriormente. Portanto, se uma chave só aparecer quando quebramos um vaso (por exemplo), seu local inicial será o zero e quando a ação "quebre o vaso" for executada, seu vetor posição será igual à posição do jogador. Não há limites de objetos por local. O nome do objeto deve ter, no máximo, 14 caracteres.

Agora vem a parte mais importante, os verbos: Pegue; Apanhe; Solte e Largue, para a manipulação de objetos (alguns verbos já estão programados no adventure por serem os mais utilizados, portanto, não é necessário defini-los).

Além dos verbos, podemos utilizar os seguintes comandos: Inventário (I), que mostra quais objetos estamos carregando; Pontos, informa sobre o escore; e Desisto, faz você recomeçar o jogo ou terminá-lo. E, é claro, Norte (N), Sul (S), Leste (L), Oeste (O), Suba e Desça, para os movimentos.

Para definir um verbo (e, portanto, uma ação), é necessário que tenhamos uma série de informações. Suponha que a descrição de um determinado local nos diz que há uma porta fechada ali. Estabelecemos que essa porta só pode ser aberta com uma chave, que deve estar com o jogador, e que ao abri-la, perdemos a chave e somos transportados para outro local qualquer. A ação será "abra a porta", e ela só pode ser executada nesse local onde há a porta e que, além disso, ao executá-la ganhamos alguns pontos.

Como codificar este exemplo? O micro vai perguntar:

1) Imperativo do verbo:

Abra.

2) Complemento:

Porta (sem utilizar artigo).

3) Resposta do micro:

"Ok. Puxa que lugar estranho!" (se quisermos que a resposta seja somente "Ok.", basta teclarmos ENTER).

4) Para que local sou transportado?

(Aqui deverá ser indicado o número do local de destino, sendo que 0 indicará a permanência no local).

5) Deslocamento de objetos:

Nesta pergunta, ao introduzirmos um número inteiro positivo, indicamos que o objeto que tem esse número irá aparecer no local. Um número inteiro negativo indica que o objeto que tem esse número (em módulo) irá desaparecer, ou melhor, sua posição será a sala zero, de que falamos há pouco. No nosso caso, se a chave for o objeto número 3, teclaremos: -3. Se o número indicado for zero, não haverá deslocamento de objetos.

6) Variação de escore (só se o jogo for por pontos):

Se o adventure for por pontos, introduziremos a variação no escore do jogador (positiva, negativa e nula) quando for executada esta ação.

7) Onde executar o comando?

Introduzindo um número inteiro positivo, indicamos que esta ação só pode ser executada no local que tenha este número. Um número inteiro negativo indica que esta ação só não pode ser executada no local que tenha este número (em módulo). Zero indica que a ação

pode ser executada em qualquer lugar.

8) Objeto necessário:

Aqui, indicamos o número do objeto necessário para a execução desta ação. O objeto deve estar em poder do jogador para que esta ação seja executada. Zero indica que não é necessário objeto algum. No nosso caso, teclaremos 3 (o número da chave).

9) Mais de uma execução?

Se a ação puder ser executada várias vezes, teclaremos S, caso contrário, teclaremos N. Se teclarmos N, o micro somente executará esta ação uma vez, recusando-se a repeti-la. Por exemplo, não podemos quebrar o mesmo vaso duas vezes.

Note que se você quiser incluir o verbo "examine", deverá programá-lo para cada objeto que puder ser examinado. O programa permite que tenhamos diversos imperativos iguais, desde que os complementos sejam diferentes; assim como permite complementos iguais, se os imperativos forem diferentes. Então, podemos ter um mesmo adventure: abra porta, abra janela, abra garrafa, quebre garrafa, tampe garrafa, examine chave, examine vaso, examine garrafa etc., desde que forneçamos as informações adequadas para cada caso.

Observações importantes: o local -1 é o que indica que o jogador morreu ou foi eliminado do jogo. Portanto, se ao executar alguma ação o jogador levar a pior, devendo ser eliminado (por exemplo: salte despenhadeiro), basta transportá-lo para o local -1. Analogamente, o local -2 é o da vitória. Quando o adventure for resolvido, basta transportar o jogador para o local -2, que automaticamente o programa desvia para uma rotina de felicitações. A passagem para estes locais pode ser feita normalmente, seja através de consequência de uma ação (ou ações) como por deslocamento voluntário (N, S, L, O, Suba ou Desça).

Além disso, quando um objeto está na posição -1 significa que está em poder do jogador, o qual poderá carregar até cinco objetos. As posições dos objetos e do jogador não interferem entre si, pois são guardadas em variáveis distintas.

Por fim, será perguntado o local inicial de jogo, e, então, serão gravadas em fita as 14 matrizes que contêm os dados introduzidos. Devido às características do micro, é necessário que apertemos uma tecla qualquer antes do SAVE, portanto, cuidado para não se distrair. Se o VERIFY acusar erros, rebobine a fita e dê um GO TO 400.

Quanto à digitação, esta não apresenta maiores problemas; as escritas em INVERSE são obtidas teclando-se INVERSE VIDEO, antes do texto, e TRUE VIDEO, depois. Cuidado máximo nas li-



## CP500/M<sup>80</sup>, uma grande decisão para homens que tomam decisões.

O CP 500/M<sup>80</sup> agora também é compatível com o CP/M, o Sistema mais difundido no mundo. Isso significa que, somando o tradicional DOS 500 com o CP/M, você passa a dispor da maior biblioteca de programas aplicativos do mercado. E você poderá tirar vantagens de programas como Calc Star\*, dBase II\*, Word Star\* e muitos outros, que vão agilizar tanto sua vida quanto a de sua empresa.

Para você, executivo, o CP 500/M<sup>80</sup> é uma poderosa ferramenta de auxílio a decisões gerenciais, com programas específicos em sua área de atuação, como análise de fluxo de caixa, controles de compras/custos/ estoques e administração financeira. Como resultado final, sua empresa terá grandes ganhos de produtividade e eficiência. CP 500/M<sup>80</sup>, um grande passo na informática com a Qualidade CP e a Tecnologia Prologica.

Características	Sistemas Operacionais Incluídos	
Memória RAM	SO-08	DOS 500
Memória ROM	64 Kbytes	48 Kbytes
Compatibilidade	2 Kbytes	16 Kbytes
Video	CP/M*	TRS-DOS
	80 colunas	64 ou 32
	por 24 linhas	colunas por
		16 linhas
Video 12" fósforo verde com controle de intensidade		
Teclado profissional capacitivo com numérico reduzido		
Interface para impressora paralela		
Opera com 1 ou 2 drives de face simples ou dupla		
Acesso a toda a Rede Nacional de Telemática, através de Interface serial padrão RS-232 C (opcional)		

\* Marcas Registradas

**CP**  
COMPUTADORES PESSOAIS  
TECNOLOGIA  
PROLOGICA



nhas 410 e 430; elas devem ser iguais, exceto pela substituição dos **SAVE** pelos **VERIFY**. Uma dica é escrever a linha 410, editá-la e substituir o número de linha e os comandos **SAVE** por **VERIFY**.

### ADVENTURE

Ao rodar o segundo programa, ele irá simplesmente perguntar o nome do adventure que desejamos carregar, e, logo após o término do carregamento, entrará automaticamente em execução.

Algumas recomendações: para movimentar-se, tecla "va para o norte" ou simplesmente **N**. Ao executar uma ação, você pode usar o artigo que quiser, tendo em mente que o programa reconhece somente a primeira e a última palavra da frase introduzida. Podemos então digitar: pegue martelo, pegue o martelo, pegue esse martelo ou pegue essa droga de martelo; que o efeito será o mesmo em todos os casos. Esse é o motivo pelo qual o Editor não aceita os artigos.

### OUTROS MACETES

Se há uma sala escura que só revela seu conteúdo se acendermos uma vela

(por exemplo), então, é conveniente termos duas salas; sendo uma com a descrição da escuridão, sem outras saídas ou objetos; e outra, que é o seu equivalente, caso tenhamos luz. Definimos assim uma ação, "acenda vela", de tal forma que só poderá ser executada na sala escura e se tivermos a vela, e que além disso, nos transporte para a outra sala, com a outra descrição. E

um verbo, "apague vela", que faça o contrário.

Experimentalmente, conseguiu-se dimensionar matrizes referentes a 100 salas, 80 objetos e 150 verbos, o que acreditamos serem suficientes para a maioria dos casos. O comando **POKE 23658,8** força o cursor para o modo **C** (maiúsculas). Se você quiser incrementar um pouco mais, experimente um **POKE 23609,30**.

### UM EXEMPLO

Aqui está um exemplo de adventure com cinco salas. Tente rodá-lo. Os números estão na ordem em que o micro os pede.

Nome - "Teste"  
É por pontos? - Não  
Quanto locais? - 5

#### LOCAL 1

Descrição: "Estamos numa velha sala de estar. Há muito pó, aqui!"

NORTE	SUL	LESTE	OESTE	SUBA	DESÇA
0	0	2	0	0	0

#### LOCAL 2

Descrição: "Estamos num corredor. Há uma escada que desce e uma porta".

NORTE	SUL	LESTE	OESTE	SUBA	DESÇA
0	0	0	1	0	4

#### LOCAL 3

Descrição: "Que lugar estranho! Há um botão na parede".

NORTE	SUL	LESTE	OESTE	SUBA	DESÇA
0	0	0	2	0	0

#### LOCAL 4

Descrição: "Está muito escuro, mas percebo uma tocha apagada na parede".

NORTE	SUL	LESTE	OESTE	SUBA	DESÇA
0	0	0	0	2	0

## Adventure

```
1) REM *****
2) REM      ADVENTURE
3) REM *****
4) REM
10) POKE 23658,8: BORDER 1: PAPER 5: INPUT "QUAL O NOME DO ADVENTURE QUE VOCE QUER JOGAR?": AS$=CLS
15) LOAD AS$ DATA Z$: LOAD "LOCALS" DATA LS$: LOAD "DESCRICAO" DATA D$: DATA M$: LOAD "OBJETOS" DATA O$: DATA V$: LOAD "VERBOS" DATA VS$: LOAD "COMPL" DATA C$: LOAD "MENSAGENS" DATA MS$: LOAD "VOCABULARIO" DATA V$: LOAD "DESCRIB" DATA D$: LOAD "PONTOS" DATA P$: LOAD "OBJETOS" DATA O$: LOAD "VERBOS" DATA VS$: LOAD "COMPL" DATA C$: LOAD "MENSAGENS" DATA MS$: LOAD "VOCABULARIO" DATA V$:
20) CLS: LET SCORE=0: LET POS=Z(1): LET SPON=Z(2): LET PONMAX=Z(3): LET NL=Z(4): LET NO=Z(5): LET NU=Z(6): DIM E(NV)
25) LET J=0: LET POSI=POS: DIM Q(NV): FOR I=1 TO NO: LET Q(I)=P(I): NEXT I: CLS
30) PRINT LS$(POS):
35) IF POS=1 THEN GOTO 470
40) IF POS=2 THEN GOTO 495
45) LET QW=0
50) FOR I=1 TO NO
55) IF P(I)<POS THEN GOTO 75
60) IF QW THEN GOTO 70
65) PRINT "OS OBJETOS VISIVEIS SAO:"; LET QW=1
70) PRINT Q(I), TO 14):
75) NEXT I
80) INPUT FLASH 1: "O QUE FAZEU AGORA?"; HS$: PRINT "": HS$=
85) IF HS$="" THEN CLS: GOTO 30
90) LET AS$="": FOR F=1 TO LEN H$
95) IF H$(F)="" THEN GOTO 105
100) LET AS$=AS$+H$(F): NEXT F
105) LET BS$="": FOR F=LEN H$ TO LEN AS$+1 STEP -1
110) IF H$(F)="" THEN GOTO 120
115) LET BS$=H$(F)+BS$: NEXT F
120) LET QW=0: FOR I=1 TO NV
125) IF AS$(I)VS$(I), TO LOCE VS$(I),
```

```
5) THEN GOTO 195
130) IF BS$(I)VS$(I), TO CODE C$(I), 5) THEN GOTO 195
135) IF V(I) AND E(I) THEN PRINT FLASH 1: "Mas ja fiz isso!": GOTO 190
140) IF B(I) THEN IF P(I)<I-1 THEN PRINT "Ainda nao posso fazer isso!": GOTO 190
145) IF H(I)=0 THEN GOTO 160
150) IF SGN U(I)=1 THEN IF POS=ABS U(I) THEN PRINT "Nao posso fazer isso aqui!": GOTO 190
155) IF U(I)>0 AND U(I)<POS THEN PRINT "Nao posso fazer isso aqui!": GOTO 190
160) PRINT "Muito bem!"
165) IF D(I) THEN LET POS=D(I): IF POS=1 THEN PRINT "Lá no pó!"
170) IF I(I)=0 THEN GOTO 185
175) IF SGN F(I)=1 THEN LET P(I)=POS: GOTO 185
180) LET PI=ABS F(I):
185) LET E(I)=1: IF SPON THEN LET T SCORE=SCORE+PI
190) LET I=NV: LET QW=1
195) NEXT I
200) IF SCORE=PONMAX THEN GOTO 495
205) IF QW THEN GOTO 35
210) IF NOT AS$="PONTOS" THEN GOTO 220
215) PRINT "TEMOS ATE AGORA "SCORE: " PONTOS": GOTO 80
220) IF NOT (AS$="PEGUE" OR AS$="A PARHE") THEN GOTO 280
225) IF BS$="" THEN PRINT "DIGA-ME PELO MENOS O QUE FAZEU": GOTO 80
230) IF J=5 THEN PRINT "Nao posso mais carregar mais nada!": GOTO 80
235) LET QW=0: FOR I=1 TO NO
240) IF NOT (BS$=O(I), TO CODE O$(I), 1) AND P(I)=POS) THEN GOTO 245
245) PRINT "OK, "AS$: " "BS$: "
250) LET QW=1
255) LET P(I)=1
260) LET J=J+1
265) NEXT I
```

```
270) IF NOT QW THEN PRINT INVERS E 1:BS$: "NAO VIJO ISSO POR AGORA!"
275) GOTO 80
280) IF NOT (AS$="SOLTE" OR AS$="LARGUE") THEN GOTO 335
285) LET QW=0
290) FOR I=1 TO NO
295) IF NOT (BS$=O(I), TO CODE O$(I), 1) AND P(I)=1) THEN GOTO 3
300) PRINT "OK, "AS$: " "BS$: "
305) LET QW=1
310) LET P(I)=POS
315) LET J=J-1
320) NEXT I
325) IF QW THEN GOTO 35
330) PRINT "Nao temos "BS$: " CONOSCO!"
335) IF AS$="DESISTO" THEN GOTO 345
340) CLS: PRINT "Neste adventure voce fez "SCORE: " pontos, isto e "SCORE/PONMAX: " % de total": GOTO 475
345) IF NOT (AS$="I" OR AS$="INVENTARIO") THEN GOTO 400
350) LET QW=0
355) FOR I=1 TO NO
360) IF P(I)<1 THEN GOTO 385
365) IF QW THEN GOTO 380
370) PRINT "Estou carregando:"
375) LET QW=1
380) PRINT TAB 10: BRIGHT 1:O(I), TO 14)
385) NEXT I
390) IF QW THEN GOTO 80
395) PRINT "Nao estou carregando nada!": GOTO 80
400) IF NOT ((AS$="N" AND M(POS,1)) OR (AS$="VA" AND BS$="NORTE" AND M(POS,1))) THEN GOTO 410
405) LET POS=M(POS,1): GOTO 30
410) IF NOT ((AS$="S" AND M(POS,2)) OR (AS$="VA" AND BS$="SUL" AND M(POS,2))) THEN GOTO 420
415) LET POS=M(POS,2): GOTO 30
420) IF NOT ((AS$="L" AND M(POS,3)) OR (AS$="VA" AND BS$="LESTE" AND M(POS,3))) THEN GOTO 430
425) LET POS=M(POS,3): GOTO 30
```

```
430) IF NOT (AS$="A" AND M(POS,4)) OR (AS$="VA" AND BS$="OESTE" AND M(POS,4))) THEN GOTO 440
435) LET POS=M(POS,4): GOTO 30
440) IF NOT (AS$="SUBA" AND M(POS,5)) OR (AS$="VA" AND BS$="SUBA" AND M(POS,5))) THEN GOTO 450
445) LET POS=M(POS,5): GOTO 30
450) IF NOT (AS$="DESCA" AND M(POS,6)) OR (AS$="VA" AND BS$="DESCA" AND M(POS,6))) THEN GOTO 460
455) LET POS=M(POS,6): GOTO 30
460) IF NOT (AS$="P" OR AS$="SIBA" OR AS$="DESCA" OR AS$="VA" THEN PRINT "Nao posso fazer isso aqui!": GOTO 80
465) PRINT FLASH 1: INK 2: PAPER 5: NAO ENTENDE A SUA ORDEM!": GOTO 80
470) PAUSE 5: BORDER 0: PAPER 0: INK 0: CLS: PRINT AT 10,0: FLASH 1: BRIGHT 1: "SINTO MUITO, MAS VOCE PERDEU... MELHOR SORTE NA PROXIMA VEZ...": FOR I=40 TO 0 STEP -1: SOUND 10,1: NEXT I
475) INPUT INK 9: "QUEM JOGAR DE NOVO?"; NV$:
480) IF NV$="" THEN GOTO 495
485) FOR I=1 TO NO: LET P(I)=Q(I): NEXT I: LET POS=POSI: LET SCORE=0: FOR I=1 TO NV: LET E(I)=0: NEXT I: LET I=0
490) BORDER 1: PAPER 5: INK 0: CLS: GOTO 30
495) PAUSE 5: BORDER 4: PAPER 6: INK 1: CLS: PRINT AT 5,0: FLASH 1: "MENS PARABENS! VOCE CONSEGUIU COMPLETAR O ADVENTURE!": FOR F=10 TO 40: SOUND 10,1: NEXT F: IF SPON THEN PRINT "VOCE FEZ "SCORE: " PONTOS, "DE UM TOTAL DE "PONMAX: " "ISIO E "SCORE/PONMAX: " % DO TOTAL": GOTO 475
```

# BRASIL TRADE CENTER

Comércio, Indústria e Participações S.A.

## COMPUTER SHOPPING

### Mesas para Microcomputadores

Fabricação própria • Cores discretas • Desenho moderno



### EQUIPAMENTOS

Micros das linhas: Sinclair • TRS-80 • CCE • Unitron • Dismac • TK 2000 II • TK 85 • Exato Pro • Impressoras • Modem • Etc.

### SUPRIMENTOS

Periféricos • Disketes • Fitas • Drives • Placas de Expansão • Interfaces • Formulário Contínuo • Cabos • Arquivos para Disketes • Etc.

### SOFTWARE

Programas e Jogos variados para todas as linhas e marcas de Computadores

### VIDEOS

Curso de Inglês em vídeo cassete • Telão • Suporte para TV • Transcodificação para todos os sistemas • Fitas: VHS - BETA-U-MATIC e para limpeza de cabeça • Baterias para 2 e 8 hs. • Iluminadores • Cabos de extensão para Câmaras • Bolsas para Câmaras e Vídeos

Nas lojas Brasil Trade Center, você também encontra vídeo-game, cartucho com jogos, som, telefonia, etc., das melhores marcas.

## VISITE UMA DE NOSSAS LOJAS E COMPROVE O QUE ANUNCIAMOS

**Matriz:** Av. Epitácio Pessoa, 280  
Ipanema-RJ — Tels.: 259-1299/259-1499

**Filiais:** Rua da Assembléia, 10 — Loja S-112  
Centro-RJ — Tel.: 222-5343

Rua Silva Vale, 416 — Cavalcante-RJ  
Tels.: 592-3047/592-3098

Rua Lopes Trovão, 134/Sala 201  
Niterói-RJ — Tel.: 710-3659

Rua Conde de Bonfim, 229 — Loja A  
Tijuca-RJ — Tel.: 284-2031

**SERVIÇO EXPRESSO REMETEMOS PARA TODO O BRASIL**

**OFERTA VÁLIDA COM A APRESENTAÇÃO DESTA ANÚNCIO**



## Editor

```

1)REM *****
2)REM **      EDITOR      **
3)REM *****
10 BRIGHT 0: PAPER 7: INK 1: B
ORDER 1: CLS : POKE 23658,0: CLS
: INPUT "QUAL E O NOME DO ADOEN-
TIVE "A$: IF A$="" THEN GO
TO 10
20 LET SPON=0: LET PONMAX=1
30 PAPER 6: CLS : PRINT "O ADV-
ENTURE E POR PONTOS ? (S/N)": I
NPUT H$: PRINT H$: IF H$(1)="N" AN-
D H$(2) S' THEN GOTO 30
40 IF H$="S" THEN LET SPON=1:
PRINT "QUAL E O SCORE MAXIMO ?":
INPUT PONMAX: PRINT PONMAX
50 PAPER 4: BRIGHT 1: CLS : IN-
PUT "QUANTOS LOCAIS TEM O ADVEN-
TIRE "I$: NL
60 DIM L$(NL,96): DIM M$(NL,6)
70 FOR I=1 TO NL
80 CLS : PRINT "O LOCAL DO L-
OCAL "I: INVERSE I: INPUT LINE
L$(I, 1 TO 96): PRINT L$(I, 1 TO 96)
90 PRINT "QUE LOCAIS ALLANCO E
XECUTANDO...":
100 PRINT "NORTE (N) "I: INP
UT H$: LET M$(I,1)=H$: PRINT H
110 PRINT "SUL (S) "I: INP
UT H$: LET M$(I,2)=H$: PRINT H
120 PRINT "LESTE (L) "I: INP
UT H$: LET M$(I,3)=H$: PRINT H
130 PRINT "OESTE (O) "I: INP
UT H$: LET M$(I,4)=H$: PRINT H
140 PRINT "SUBA "I: INP
UT H$: LET M$(I,5)=H$: PRINT H
150 PRINT "DESCA "I: INF

```

```

UT H: LET M(I,6)=H: PRINT H
160 INPUT "TUOO OK? "H$: IF H$
="N" THEN GOTO B0
170 NEXT I
180 PAPER 1: BRIGHT 1: INK 7: C
LS : INPUT "QUANTOS OBJETOS HA N
O JOGO ? "N0: DIM O$(N0,15): DI
M P(N0)
190 FOR I=1 TO N0
200 CLS : PRINT "QUAL O OBJETO
": INVERSE I: INPUT H$: LET O
$(I, 1 TO 14)=H$: LET O$(I,15)=CHR
$(LEN H$+1) PRINT H$
210 PRINT "EM QUE LOCAL SE ENCO
NTRA? "I: INPUT P(I): PRINT P(I)
220 INPUT "TUOO OK? "H$: IF H$
="N" THEN GOTO 200
230 NEXT I
240 PAPER 2: INPUT "QUANTOS VER
BOS? "N1: DIM V$(N1,15): DIM C$
(N1,15): DIM H$(N1,64): DIM D(N1
): DIM B(N1): DIM A(N1): DIM U(N
): DIM B(N1): DIM K(N1): FOR I=
1 TO N1
250 CLS : PRINT "QUAL O VERBO "
I: INPUT H$: LET V$(I, 1 TO 14)=
H$: LET V$(I,15)=CHR$(LEN H$+1)
PRINT H$
260 PRINT "QUAL SEU COMPLEMENT
O ? "I: INPUT H$: LET C$(I, 1 TO 14
)=H$: LET C$(I,15)=CHR$(LEN H$+1)
PRINT H$
270 PRINT "QUAL A RESPOSTA OD
MICRO ? "I: INPUT LINE H$: IF H$="
" THEN LET H$="OK."
280 LET H$(I, 1 TO 15)=H$: PRINT H$
(I, 1 TO 15)
290 PRINT "PARA QUE LOCAL SOU
TRANSPORTADO (0=NAO SOU TRASNPO
RTADO "I: INPUT O(I): PRIN
T O(I)
300 PRINT "O DESLOCAMENTO DE OBJ
ETO? "I: INPUT F(I)
310 IF SPON THEN PRINT "VARIAC
AO DE SCORE? "I: INPUT F(I)
320 PRINT "ONOE POSSO EXECUTAR
O COMANDO? "I: INPUT U(I): PRIN
T "SALA "I: U(I)
330 PRINT "QUE OBJETO DEVO TER
? (0=NINGUÉM) "I: INPUT B(I): IF B
(I) THEN PRINT O$(B(I), 1 TO CODE
O$(B(I),15))
340 PRINT "POSSO EXECUTAR ESTE
VERBO MAIS DE UMA VEZ? (S/N)":
INPUT H$: IF H$="N" THEN LET K(I
)=1: GOTO 360
350 LET K(I)=0
360 INPUT FLASH 1: "TUOO OK? "I:
H$: IF H$="N" THEN GOTO 250
370 NEXT I
380 PAPER 4: INK 0: BRIGHT 0: C
LS : PRINT "QUAL O LOCAL DE INIC
IO? "I: INPUT POS: PRINT POS
390 DIM Z(6): LET Z(1)=POS: LET
Z(2)=SPON: LET Z(3)=PONMAX: LET
Z(4)=NL: I: Z(5)=N0: LET Z(6)=
NV
400 PAPER 7: INK 2: CLS : PRINT
AT 10,5: "VOU GRAVAR AS MATRIZES
APORTE UMA TECLA": P
AUSE 0
410 SAVE A$ DATA Z(I): SAVE "LOC
(I, 1 TO 15)

```

## LOCAL 5

Descrição: "Estamos num porão. Veja uma saída ao Sul e uma caixa no chão."

NORTE	SUL	LESTE	OESTE	SUBA	DESÇA
0	1	0	0	2	0

Quantos objetos? - 7

OBJETO	DESCRIÇÃO	LOCAL
1	CHAVE	0
2	ISQUEIRO	1
3	MAPA	1
4	RELOGIO	2
5	MICRO	3
6	MARTELO	5
7	SERROTE	3

Número de verbos - 6

Para exemplificar a entrada dos verbos, complementos e outras informações sobre o adventure, descreveremos apenas o primeiro exemplo. Portanto, valerão para todos os exemplos, após o primeiro, as mesmas perguntas.

**Verbo 1 - Abra**  
Complemento - Caixa  
Resposta do micro - Veja o que achamos!  
Transportado para - 0  
Deslocamento de objetos - 1  
Onde pode executar - 5  
Objeto que deve ter - 0  
Pode-se executar mais de uma vez - N

**Verbo 2**  
Aperte  
Botão  
Ok (tecle apenas Enter)  
-2, 0, 3, 0, N

**Verbo 3**  
Abra  
Porta  
Ok  
3, 0, 2, 1, N

**Verbo 4**  
Examine  
Mapa  
É um mapa de Hong Kong!  
0, 0, -1, 3, S

**Verbo 5**  
Acenda  
Tocha  
Assim é melhor  
5, 0, 4, 2, S

**Verbo 6**  
Apague  
Tocha  
Ok  
4, 0, 5, 0, S

Local de início: -1

## VARIÁVEIS UTILIZADAS PELOS PROGRAMAS

**L\$** (matriz alfanumérica) - Contém as descrições dos vários locais.

**M** (matriz numérica) - Contém os vetores de deslocamento nas seis direções (N, S, L, O, Suba e Desça).

**O\$** (matriz alfanumérica) - Contém os nomes dos objetos, sem artigo.

**P** (matriz numérica) - Contém a posição de cada objeto. Há posições particulares!

-1 indica que o objeto está com o jogador

0 indica que o objeto está no local zero (guarda-objetos)

**V\$** (matriz alfanumérica) - Contém o imperativo dos verbos do jogo.

**C\$** (matriz alfanumérica) - Contém o complemento de cada verbo.

**M\$** (matriz alfanumérica) - Contém a mensagem a ser dada ao jogador quando for executado o verbo correspondente.

**D** (matriz numérica) - Contém o eventual deslocamento do jogador relativo ao verbo correspondente.

**F** (matriz numérica) - Contém o eventual deslocamento de objetos (aparecimento ou desaparecimento) relativo à execução do verbo correspondente.

**A** (matriz numérica) - Contém os pontos relativos à execução do verbo correspondente.

**U** (matriz numérica) - Contém o flag que indica onde o verbo correspondente pode ser executado (0 = em qualquer lugar).

**B** (matriz numérica) - Contém o número do objeto necessário à execução do verbo correspondente.

**K** (matriz numérica) - Contém o flag que indica se determinado verbo pode ser executado mais de uma vez.

**Z** (matriz numérica) - Contém os seguintes flags:

**Z(1)** = posição inicial  
**Z(2)** = flag que indica se o adventure é por pontos  
**Z(3)** = escore a ser alcançado  
**Z(4)** = número de locais  
**Z(5)** = número de objetos  
**Z(6)** = número de verbos

**J** - Número de objetos em poder do jogador

**A\$** - Primeira palavra introduzida

**B\$** - Última palavra introduzida

**PONMAX** - Escore a ser alcançado

**SPON** - Indica se o adventure é por pontos

**SCORE** - Escore do jogador

**POS** - Posição do jogador (ver texto)

**H\$** - Ordem do jogador

Paolo Fabrizio Pugno cursa atualmente o primeiro ano da Faculdade de Engenharia Industrial (SP) e possui um ZX Spectrum, com o qual desenvolve programas.



## CONTINUAMOS RENOVANDO

A mais completa linha de Software para o Color Computer CP. 400, Color 64, MX-1600

MICROMAQ - Rua Sete de Setembro, 92 - Lj. 106 - Tel.: (021) 222-6088 CEP 20050 RJ

## PROMOÇÃO DE FÉRIAS

Fitas com dois programas

Seleção de alguns dos melhores "games" agrupados dois a dois.

PR.001 - Zaxxon/Buzzard bait  
PR.002 - Draconian/8 ball  
PR.003 - Speed racer/Bag-man  
PR.004 - Galagon/Demon seed  
PR.005 - Space wreck/Sea dragon  
PR.006 - Sanctum/R-ball  
PR.007 - Sea quest/Quix  
PR.008 - Calixto island/Pooyan

PR.009 - Lunar rover/Devil assault  
PR.010 - Time patrol/Moon shuttle  
PR.011 - Canyon climb/Fly tiger  
PR.012 - Decathlon/Cashman  
PR.013 - Color car/Kron  
PR.014 - Rat/The king  
PR.015 - Cristal/Doodle bug  
PR.016 - Time bandit's/ Tut's tomb

PR.017 - Time fighter/Moon hopper  
PR.018 - Mudpies/Death trap  
PR.019 - Crash/Ranger  
PR.020 - Color panik/Shenanigans

ATENÇÃO: qualquer fita acima 40.000.

as fitas acima não sofrem desconto de qualquer espécie.

Solicite nosso catálogo com mais de 300 produtos

## RECORDISTAS DE VENDAGEM - 1985

E006	WORLD'S OF FLIGHT - Simulador de voo	50.000	PD001	CALIXTO - Ajuda o arqueólogo (Prof. Lagarto) a recuperar o tesouro	35.000
PP002	CYRUS - Para amantes do jogo de xadrez	35.000	PD003	SEA QUEST - Recupere o tesouro perdido	35.000
PJ011	ZAXXON - Enfrente os perigos de uma plataforma espacial	35.000	PD004	BLACK SANCTUM - Exorcize o demônio	35.000
PJ005	CRISTAL - Sensacional - estilo Pac-Man em 3D	35.000	PJ023	DEMON SEED - Destrua pássaros, sementes e naves	35.000
E006	SAIOL - Derrote o Brutus e conquiste o coração da Olívia. 3 telas. 9 níveis de dificuldade	35.000	E002F	MUSICA 2 - Para compor, arranjar, copiar partituras etc.	140.000
			PJ008F	SUPER SCREEN - Aumenta o tamanho da tela (52 colunas, 24 linhas). Acrescenta funções diversas	100.000
PJ051	SPEED RACER - Corrida de carros (tipo flipper)	35.000	PA002F	ELITE CALC - Planilha eletrônica. Até 255 linhas e colunas. Manipula textos, números etc. Emite gráficos e permite ordenar colunas e linhas. Manual em português	150.000
PJ039	DRACONIA - Salve os prisioneiros do espaço e fuja do dragão	35.000	PL001F	EDTASM - Linguagem Assembler	150.000
PJ018	BUZZARD BAIT - Ataque os pássaros com sua lança	35.000	PL005F	COLOR LOGO - Linguagem educativa logo	180.000
PJ065	GALAGON - Idêntico ao Fantástico do flipper	35.000	PA007F	PRO COLOR FILE - O melhor banco de dados já criado para o color. Somente em disco (preço incluso)	270.000
PJ058	QUIX - Idêntico ao do flipper. Preencha 75% da tela sem ser atingido	35.000	PA006F	TIMS/SIGA - Banco de dados que permite criar até 8 campos. Manual em port.	130.000
PJ059	DECATHLON - 10 provas olímpicas	35.000			
PJ071	SPACE WRECK - Defenda sua base espacial do ataque inimigo. 3D com naves estilizadas	35.000			

Para programas em disco, acrescer 60.000 por programa (Disco + Embalagem)

## A GRANDE SENSACÃO

**COLOR MAX** (módulo de alta-resolução) - Reproduz no color com 64K a performance gráfica apresentada pelo Macintosh, imitando as funções do MacPaint. Utiliza técnicas modernas de programação tipo "Icons" e "Pulldown Menus". Composto de placa de Hardware e programa em fita ou disco

(Para cálculo de ORTN, considerar a data da postagem)

E001F 14-ORTN CONSULTE.  
E001D 15-ORTN BAIXOU!

## OUTROS PROGRAMAS

• JOGOS SÓ PARA DISCO (preço c/disco incluso)			• APLICATIVOS P/FITA		
PS002D	EAGLE - Simula um voo espacial em que você deve aportar e levantar voo da lua ou de marte. Manual em Inglês. Requer 2 Joysticks	90.000	PA001F	WRITER II - Tela de até 85 colunas, "Window" do texto com a forma final. Permite inserção de caracteres em português (na imp.). Mais de 100 funções	200.000
PS003D	THE FACTORY - Crie a linha de montagem de uma fábrica e teste a sua memória	90.000	PA013F	MALA POSTAL (I) - Totalmente criado no Brasil. Emite etiqueta, relatório e lista telefônica	160.000
PD005D	SANDS OF EGYPT - Descubrir o mistério da tumba do Faraó	90.000	PA017F	ASTROLOGIA - Gerador de mapas astrológicos	150.000
PD008D	CRIME - Desvende o mistério de um crime. Afinal, você é o detetive	90.000	PA030F	HANDY MAN - Auxiliar no cálculo de custos de obras civis, cálculo de áreas, coberturas etc.	35.000
PD009D	MAJOR ISTAR - Viajando através das cidades submarinas do século 21, você deverá desvendar o mistério de Trident	90.000	PA021F	STATGRAF - Estatística. Gera gráficos com regressão linear, médias, desvios etc.	160.000
PD010D	SAM SLEUTH - Desvende o mistério de um desaparecimento	90.000			
PD011D	STAR TRADER - Mercador das galáxias	90.000			
• GRÁFICOS			• APLICATIVOS P/DISCO		
ART GRAF	Criado pela Micromaq. Necessita 64K. Trabalha com 8 telas simultâneas na memória no mais alto modo gráfico.	200.000	PA001D	VIP WRITER - Idem ao PA001F	260.000
PG004F		260.000	PA015D	VIP DATABASE - Banco de dados	260.000
PG004D			PA029D	MATHMENU - Sistema para auxílio no estudo da matemática	220.000
VIZI DRAW	Muito potente para desenhar. Acompanha o Vizidump, que leva suas telas para a impressora	300.000	PA011D	DYNACALC - Planilha eletrônica. Gera gráficos em alta resolução	260.000
ART GALLERY	Potente gerador de gráficos e telas de desenho	150.000	PA028D	ELCIRCAN - Analisador de circuitos elétricos	220.000
PG005D			PA021D	STATGRAF - Idem ao PA021F	220.000
PG002F					
• SOM, SINTETIZAÇÃO			• UTILITÁRIOS EM FITA		
COLOR OUAVER	Para compor, arranjar, copiar partituras, monta espectro, cria ondas etc.	160.000	PU013F	VDOS - Simula o sistema operacional de disco em fita	120.000
PM003F		220.000	PU014F	FAST TAPE - Dobra a velocidade de leitura e gravação para cassete	130.000
PM003D			PU019F	COLOR KIT - Ferramenta muito potente para trabalho com a memória. Acrescenta mais 30 funções ao Color.	160.000
• EDUCATIVOS			• UTILITÁRIOS EM DISCO		
TABELA PERIÓDICA DE QUÍMICA		40.000	PU001D	ADOS - Sistema de DOS. O mais utilizado atualmente. Cria novas funções	220.000
PE012F		100.000	E003D	CBASIC COMPILER - O melhor compilador até o momento. Compila tudo	320.000
PE012D		40.000	PU019D	COLOR KIT - Idem PU019F	220.000
PE013F		100.000	PU020D	VIP DISK ZAP - Altera todo o conteúdo de um disco	320.000
PE013D		100.000	PU018D	WHERE IS IT? - Gerenciador de diretórios. Auto-explicativo	150.000

## Desconto

até 150.000 sem desconto  
de 151.000 a 250.000 5%  
de 251.000 acima 10%

Desejo receber o(s) seguinte(s) programa(s), pelo(s) qual(is) pagarei a quantia de Cr\$

PROGRAMAS:

NOME:

END:

CIDADE:

UF:

CEP:

Para tal, estou enviando ( ) cheque ( ) Vale Postal à MICROMAQ LTDA.

Rua Sete de Setembro, 92 - Lj. 106 - CEP 20050 - Rio de Janeiro - RJ.

Despesas de correio incluídas.

ATENÇÃO: Só serão aceitos os pedidos com postagem até 31/01/86.





Gilson Roberto Viana

# Thunderbird

**Thunderbird** é um programa animado, baseado numa batalha aérea onde você tem, como árdua tarefa, a missão de pilotar um avião de bombardeio por uma região montanhosa e infestada de mísseis antiaéreos e caças inimigos. Cada projétil inimigo abatido por um tiro ou uma bomba aumentará sua pontuação em 10 pontos.

Cada vez que você colidir com um obstáculo (montanha, míssil ou caça), perderá uma nave e o jogo só se reiniciará quando for pressionada a tecla **NEWLINE**. O jogo acaba quando você perder suas três naves.

A cada reinício de partida, o leitor pode modificar a velocidade de acordo com o nível escolhido, que pode ser lento (1), médio (2) e rápido (3). Entretanto, se você estiver realizando uma competição, é aconselhável manter o mesmo nível, para que todos os jogadores pilotem com o mesmo grau de dificuldade.

Caso você consiga quebrar o recorde antigo, terá o privilégio de escrever seu nome no quadro dos campeões. O nome pode ter, no máximo, sete letras e, se for maior, deve ser completado com pontos.

Os controles podem ser usados simultaneamente. São eles:

- 1) Digite a listagem 1;
- 2) Digite **GOTO 100** e **NEWLINE**;
- 3) Quando for pedido endereço inicial, digite **16514** e **NEWLINE**;
- 4) Digite os códigos em linguagem de máquina da listagem 2 e, quando acabar digite "...";
- 5) Quando, novamente for pedido endereço inicial, digite **30720** e **NEWLINE**;
- 6) Digite os códigos em linguagem de máquina da listagem 3 e, quando terminar, digite "XX";
- 7) Apague (delete) as linhas de 100 até 210;
- 8) Finalmente, se tudo estiver correto, prepare a fita para gravar e digite **RUN** e **NEWLINE**.

Figura 1

6 — desce; 7 — sobe; 9 — dispara bombas (máximo de três bombas) e 0 — dispara tiros (máximo de cinco tiros).

Veja, na figura 1, como você deve proceder na digitação do programa.

*Gilson Roberto Viana é estudante do Curso de Engenharia Elétrica, na UFPR, e Programador autodidata há dois anos.*

```
1 REM 11111111111111111111
111111111
10 DIM K$(110)
20 RAND USR 16528
30 SAVE "THUNDERBIRD"
40 RAND USR 16535
100 CLS
105 PRINT "ENDEREÇO INICIAL ?"
:
107 INPUT N
110 PRINT N
115 LET B=N
120 LET A$=""
125 IF A$(2 TO 7)="" THEN INPUT
A$
130 IF A$(2 TO 7)="" THEN GOTO 1
25
135 GOTO (A$(1)-1)*(-40)+(A$(7)-
X")*(-50)+140
140 POKE N,(CODE A$-28)*16+CODE
A$(2)-28
```

Listagem 1

```
16514 2A 10 40 11 00 00 19 11
16522 00 78 01 55 04 C9 CD 82
16530 40 EB ED 80 C9 CD 82 40
16538 ED 80 C3 88 7B 00 00 00
```

Listagem 2

```
30720 80 80 80 80 80 80 80 97
30728 97 80 80 AD BA B3 A9 AA
30736 B7 A7 AE B7 A9 80 97 97
30744 80 80 80 80 80 80 80 80
30752 80 00 2C 2E 31 38 34 33
30760 00 38 2E 26 33 26 00 16
30768 00 32 2E 28 37 34 00 38
30776 2E 38 39 2A 32 26 38 80
30784 07 03 84 B2 A6 BD AE B2
30792 B4 80 AF B4 AC A6 A9 B4
30800 B7 80 80 B5 B1 A6 AB A6
30808 B7 80 B9 AA B2 B5 B4 B0
30816 05 1F 85 1C 1C 1C 1C 1C
30824 1C 00 0F 0F 0F 0F 0F 0F
30832 0F 00 80 1C 1C 1C 1C 1C
30840 1C 80 1C 1E 25 25 25 80
30848 01 01 5B 41 54 41 A3 0B
30856 03 0B 0B 03 03 03 0B 03
30864 0B 03 0B 03 0B 03 0B 03
30872 E5 D5 C5 2A 0C 40 11 EE
30880 02 19 5F 7E FE 25 20 05
```

## REPRESENTANTES AUTORIZADOS

Para sua maior comodidade, a ATI Editora Ltda. coloca à sua disposição os seguintes endereços de seus representantes autorizados

**BELO HORIZONTE**  
Maria Fernanda G. Andrade  
Caixa Postal 1687  
Tel.: (031) 335-6645



**RIO DE JANEIRO**  
ATI Editora Ltda.  
Av. Pres. Wilson, 165 - gr. 1210  
CEP 20030 - Tel.: (021) 262-6306

**SÃO PAULO**  
Rua Oliveira Dias, 153  
Jardim Paulista  
CEP 01433 - Tel.: (011) 853-3574

**PORTO ALEGRE**  
Com. Rep. Odilon Ltda.  
Rua Vol. da Pátria, 323 - Cj. 309  
CEP 90000 - Tel.: (0512) 24-8200 R. 309

**Micro  
Sistemas**

```
30888 36 1C 2B 18 F6 FE 80 28
30896 01 34 7B C1 D1 E1 C9 2A
30904 0C 40 01 F5 02 09 3E 1C
30912 BE 20 04 2B BE 28 FC 3E
30920 80 BE 28 09 35 23 BE C8
30928 36 25 18 F9 E1 2A 84 78
30936 16 08 E5 06 12 7E FE 76
30944 28 06 C6 80 77 23 18 F5
30952 23 10 F2 CD FD 7A E1 15
30960 20 E8 C9 AF D8 FE C8 47
30968 C0 21 80 78 7E FE 05 C8
30976 34 2A 82 78 23 97 BE 20
30984 03 36 16 C9 C8 66 28 04
30992 77 CD 98 78 C3 85 79 AF
31000 D8 FE C8 4F C0 21 81 78
31008 7E FE 03 C8 34 2A 82 78
31016 23 11 21 00 19 3E 87 BE
31024 CA 8C 79 C3 4C 79 2A 82
31032 78 23 11 21 00 19 7E FE
31040 8A C8 FE 87 20 F7 36 00
31048 19 BE 28 FC 97 BE 20 03
31056 36 87 C9 C8 46 28 04 77
31064 CD 98 78 C3 8C 79 2A 84
31072 78 23 7E FE 8A C8 FE 16
31080 20 F7 36 00 23 BE 28 FC
31088 97 BE 20 04 36 16 18 E9
31096 CD 92 79 C8 46 28 E2 77
31104 CD 98 78 18 DC FE 16 28
31112 09 FE 87 C0 E5 21 81 78
31120 18 04 E5 21 80 78 35 E1
31128 C9 21 86 78 34 7E 21 00
31136 09 23 3D 20 FC 23 3E 55
31144 B6 4F 3E 0E A6 C8 5F 28
31152 F4 2A 0C 40 11 92 02 19
31160 11 21 00 36 08 ED 52 3D
31168 20 F9 79 FE 7F 28 06 FE
31176 FF C0 36 13 C9 36 2D C9
31184 23 BE 28 28 02 36 00 28
31192 BE 20 09 23 23 BE CA 78
31200 7A 28 18 0C 7E FE 82 CA
```

```
31208 D4 78 CD 85 79 36 08 23
31216 C3 9C 7A 36 00 E5 09 7E
31224 CD 26 7A E1 2B FE 08 28
31232 03 AF ED 42 7E FE 03 28
31240 06 FE 00 20 06 36 2D 23
31248 C3 81 7A FE 86 28 02 FE
31256 82 CA D4 78 36 00 CD 98
31264 78 CD 85 79 18 E9 D5 C5
31272 5F 23 16 07 03 AF ED 42
31280 7E FE 86 28 04 15 20 F5
31288 78 C1 D1 C9 36 00 1A FE
31296 08 28 02 09 09 AF ED 42
31304 7E FE 08 20 03 12 18 F5
31312 FE 03 20 03 12 18 EC 28
31320 7E FE 00 C2 13 7A 36 13
31328 13 23 C3 9C 7A 2A 84 78
31336 23 11 87 78 01 21 00 E5
31344 3E 0C 36 00 09 3D 20 FA
31352 E1 23 7E FE 00 CA 9F 7A
31360 E5 7E FE 8A 20 03 E1 18
31368 F0 FE 00 28 0F FE 08 CA
31376 00 79 FE 2D CA F3 79 FE
31384 13 CA 3C 7A 09 18 E2 CD
31392 99 79 CD 5E 79 CD 36 79
31400 18 11 2A 0C 40 11 64 00
31408 19 22 84 78 11 07 00 19
31416 97 18 27 2A 82 78 E5 D1
31424 01 21 00 AF D8 FE C8 5F
31432 28 06 C8 67 20 20 09 09
31440 ED 42 7E FE 00 28 07 FE
31448 03 28 13 C3 D5 78 97 12
31456 18 12 36 82 28 BE 20 F3
31464 36 86 23 22 82 78 CD F3
31472 78 CD 17 79 CD B7 78 CD
31480 FD 7A C3 65 7A 01 01 0F
31488 08 79 80 20 FB C9 2A 0C
31496 40 11 DA 02 19 EB 21 10
31504 00 19 C9 CD 06 78 13 23
31512 1A FE 00 C8 BE 28 F7 96
31520 D0 CD 06 78 01 06 00 ED
```

```
31528 B0 D5 21 69 78 01 08 00
31536 ED 80 E1 23 36 04 CD 48
31544 7B D6 1B 38 F9 D6 25 30
31552 F5 C6 40 77 23 7E FE 00
31560 C8 18 E9 E5 D5 CD 88 02
31568 E5 C1 2C 28 F8 CD BD 07
31576 7E F5 CD 88 02 2C 20 FA
31584 F1 D1 E1 C9 2A 0C 40 11
31592 65 00 19 06 11 36 00 23
31600 7E FE 80 20 F8 23 23 10
31608 F6 C9 36 80 06 1E 23 77
31616 10 FC 23 36 80 23 23 C9
31624 ED 5B 0C 40 13 21 00 78
31632 01 20 00 C5 ED 80 C1 13
31640 ED 80 EB 23 3E 03 CD 7A
31648 7B 3E 11 01 1F 00 36 80
31656 09 36 80 23 23 3D 20 F6
31664 3E 8A CD 7A 78 EB 01 20
31672 00 C5 ED 80 C1 13 ED 80
31680 EB 23 36 82 23 36 83 23
31688 36 81 06 1D 23 36 80 10
31696 FB CD F7 78 CD 48 78 FE
31704 76 20 F9 CD 64 78 21 80
31712 78 97 77 23 77 CD AA 7A
31720 2A 0C 40 11 D8 02 19 7E
31728 FE 1D 28 47 35 18 DD CD
31736 64 78 2A 0C 40 01 0E 01
31744 09 11 2D 7C EB 05 ED 80
31752 21 26 7C CD 48 78 01 03
31760 00 ED B1 20 F3 CB 11 09
31768 4E 21 FF 7A 71 EB 28 BE
31776 20 FC C6 80 77 C9 1D 1E
31784 1F 00 00 09 0F 33 2E 3E
31792 2A 31 0E 00 1D 00 00 18
31800 00 00 1F E5 CD 13 7B E1
31808 36 1F CD 06 78 11 73 78
31816 EB 01 0C 00 ED 80 C3 D1
31824 78 90 00 00 00 00
```

Listagem 3

## COMPUTAÇÃO É COM A CAMPUS

### BEST SELLERS:

### LANÇAMENTOS

### JANEIRO/86

- 1 MUMPS — Lobo, M. C. — Cr\$ 49.000
- 2 Inteligência Artificial em BASIC — James, M. — Cr\$ 90.000
- 3 VISITREND/VISIPLOT: Guia do Usuário Brasileiro — Christmann, R. U. — Cr\$ 54.000
- 4 LOGO: Uma Introdução — Goodyear, P. M. — Cr\$ 82.000
- 5 Linguagens de Programação para Micros — Marshall, G. — Cr\$ 72.000
- 6 Pascal para Micros — James, M. — Cr\$ 68.000
- 7 Técnica de Gerenciamento de Arquivos — Claybrook, B. G. — Cr\$ 125.000

### APPLE

- 8 Como Programar seu APPLE — Cook, R. e Hartnell, T. — Cr\$ 41.000
- 9 Jogos Gráficos para o APPLE — Coletta, P. — Cr\$ 82.000
- 10 Manual APPLE em BASIC — Kanteris, N. — Cr\$ 52.000

### BASIC

- 11 BASIC para Aplicações Comerciais — Hergert, D. — Cr\$ 55.000
- 12 BASIC com Estilo — Nagin, P. e Ledgard, H. F. — Cr\$ 48.000
- 13 BASIC Básico — Pereira F9, J. C. — (5ª edição revisada e ampliada) — Cr\$ 64.000

### LINGUAGEM DE MÁQUINA

- 14 Além do BASIC — Linguagem ASSEMBLY para a Linha SINCLAIR — Santos, N. — Cr\$ 60.000

- 15 BASIC Rápido: Além do BASIC TRS-80 — Gratzner, E. A. e Gratzner, T. G. — Cr\$ 99.000
- 16 Manual do Microprocessador Z-80 — Barden Jr., W. — Cr\$ 109.000
- 17 Programação em ASSEMBLER e Linguagem de Máquina (2ª edição) — Alexander, D. C. — Cr\$ 68.000
- 18 Desafio: Os Mais Excitantes Jogos em BASIC — Boteinho, A. J. L. — Cr\$ 87.000
- 19 Primeiros Passos com o seu SINCLAIR (TKs) — Hughes, C. — Cr\$ 49.000
- 20 Programas Administrativos em BASIC SINCLAIR — Karsten, L. — Cr\$ 57.000
- 21 30 Programas em BASIC para Computadores Pessoais — Chance, D. — Cr\$ 92.000
- 22 1001 Aplicações para seu Computador Pessoal — Sawusch, M. — Cr\$ 86.000

ESTA É APENAS UMA SELEÇÃO DE NOSSOS TÍTULOS. PARA MAIORES INFORMAÇÕES SOLICITE NOSSO CATÁLOGO.

**Editora Campus**  
Rua Barão de Itapagipe 55 - 20261 - RJ - RJ - Tel.: (021) 284 8443

**SIM** DESEJO RECEBER AS PUBLICAÇÕES REFERENTES AOS NÚMEROS ASSINALADOS:

1	2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21	22

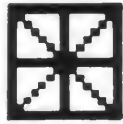
Nome: .....  
Endereço: .....  
Cidade: ..... CEP: ..... Estado: .....

- ☐ Mediante envio de cheque nominal à Editora Campus Ltda. (O porte é por nossa conta e sua encomenda chega mais rápido.)  
Cheque nº: .....  
Banco: .....
- ☐ Mediante pagamento contra recebimento dos livros pelo reembolso postal, acrescido das despesas postais.

Importante: Preços válidos até 28.02.86. Após esta data informar se concorda com o envio do pedido com reajuste.  
☐ SIM ☐ NÃO

NOSSAS PUBLICAÇÕES ESTÃO A VENDA TAMBÉM NAS BOAS LIVRARIAS





José Marcelo de Freitas Vilela

O programa "FUGA/BAS" é um jogo rodado em BASIC, mas que possui uma pequena rotina em linguagem de máquina, nas linhas 20 e 21. Esta rotina ativa a unidade sonora do equipamento, através da instrução J = USR(X); sendo J uma variável muda e X a frequência e a duração do som desejado.

O programa foi desenvolvido para os micros que têm 48 Kb de memória, compatíveis com os modelos I e III do TRS-80 (quer possam ou não unidades de disco).

A listagem apresentam-se de forma extensiva porque no programa estão contidas muitas sub-rotinas gráficas, além da presença de sons, ruídos e trechos de músicas.

### O JOGO

O jogo consiste numa fuga, na qual o jogador é um guerreiro espacial que precisa sair o mais rápido possível (20 minutos é o tempo máximo) de uma Zona Proibida (ZP), que é

**N** — Ir para o norte  
**S** — Ir para o sul  
**L** — Ir para o leste  
**O** — Ir para o oeste  
**M** — Ver o mapa (caso você o possua)  
**T** — Ver o tempo que resta  
**E** — Entrar na passagem  
**R** — Ver quantos diamantes restam

Figura 1

dividida entre Zona Proibida 1 e Zona Proibida 2. O jogador terá vários comandos, conforme mostra a figura 1.

O jogador precisa ter cuidado com os ciclones espaciais, com o tempo e nunca perder todos os diamantes de Orion, pois eles são o combustível da nave; perdendo-os, acaba-se o jogo. Cada lugar da Zona Proibida é chamado quadrante, isto é, uma espécie de sala no espaço sideral.

Quando se acaba o jogo (tenha o jogador

ganho ou não), o computador faz uma conta com o número de jogadas e os diamantes conseguidos, dando o seu escore.

Quando o jogador estiver voando com sua nave na ZP, a tela do equipamento irá se subdividir em três partes: na parte de baixo, está o menu; no canto superior esquerdo, o computador de bordo; e no canto superior direito, o visor da nave.

Durante a fuga cósmica, o jogador deve se preocupar em achar o Mapa e a Estrela do Poder, pois possibilitarão surpresas especiais.

Agora, digite o programa com cuidado e atenção; ao acabar, digite RUN e tecla ENTER que o computador dirá tudo. Boa Sorte!

*José Marcelo de Freitas Vilela tem 14 anos e cursa a oitava série no Colégio São Bento, no Rio de Janeiro. Possui um CP500 e aprendeu a programar em BASIC através de livros e revistas.*

# Fuga cósmica

```
0 CLEAR:J=0:POKE16419,95
1
2 ' FUGA COSMICA P/COMP. TRS-80 MOO.1 e III
3 ' DE JOSE MARCELO DE FREITAS VILELA
4 ' Rio de Janeiro / 1985 / TELEF.1(021)270 4300
5
6 CLS:FORAA=1:OT29:FORBB=1:OT29:NEXTT88:REAOC:REAOL:SET(C,L):NEXTAA
7 OATA 10,10,11,10,10,17,11,17,10,16,11,16,10,15,11,15,10,14,11,14,10,13
8 OATA 11,13,10,12,11,12,10,11,11,11,10,10,11,10,12,9,13,9,14,9,15,8,16,8,17,8
9 OATA 18,8,19,9,20,9,12,14,13,14,14,15,14
10 DATA 20,14,20,15,20,16,20,17,21,18,22,18,23,18,24,18,25,18,26,17,26,16
11 DATA 26,15,26,14
12 OATA 35,14,34,14,33,14,32,1,31,14,30,15,30,16,30,17,31,18,32,18,33,18
13 DATA 34,18,35,18,36,17,36,16,36,15,36,18,36,19,36,20,36,21,35,22,34,22,33,22
14 DATA 32,22,31,22,30,21
15 OATA 45,14,44,14,43,14,42,14,41,14,40,15,40,16,40,17,41,18,42,18,43,1
16 DATA 8,44,18,45,18,46,17,46,16,46,15,46,18,47,18
17 OATA 53,38,52,38,52,39,51,39,50,39,49,39,48,39,47,39,46,39,45,39,45,3
18 DATA 8,44,38,44,37,43,37,44,36,43,36,44,35,43,35,44,34,43,34,43,33,44,33,43,3
19 DATA 2,44,32,43,31,44,31,43,30,44,30,44,29,45,29,45,28,46,28,47,28,48,28,49,2
20 DATA 8,50,28,51,28,52,28
21 OATA 52,29,53,29,61,34,60,34,59,34,58,34,57,34,56,35,56,36,37,56,3
22 DATA 8,57,39,58,39,59,39,60,39,61,39,62,38,62,37,62,36,62,35,60,33,61,32
23 DATA 14, OATA 70,34,69,34,68,34,67,34,66,34,65,35,66,36,67,36,68,36,69,36,70,3
24 DATA 6,71,37,71,38,70,39,69,39,68,39,67,39,66,39
25 OATA 74,39,74,38,74,37,74,36,74,35,75,34,76,34,77,34,78,35,78,36,78,3
26 DATA 7,78,38,78,39,79,34,80,34,81,34,82,35,82,36,82,37,82,38,82,39
27 OATA 16, OATA 85,39,85,38,85,37,85,36,85,35,86,35,85,34,86,35,85,33,86,34,92,34,91,34,90,3
28 DATA 4,89,34,88,35,88,36,88,37,88,38,89,39,90,39,91,39,92,39,93,39,94,38
29 DATA 17, OATA 102,34,101,34,100,34,99,34,98,34,97,35,97,36,97,37,97,38,98,39,9
30 DATA 9,39,100,39,101,39,102,39,103,38,103,37,103,36,103,35,103,39,104,39
31 DATA 18, FORA=1:OT29:FORB=1:OT30:NEXTT88:REAOC:POKE16277+A,C:NEXTA
32 OATA 68,101,0,74,111,115,101,39,0,77,97,114,99,101,108,111,0,100,101,
33 DATA 0,70,40,0,86,105,108,101,108,97
34 DATA 20, Z=STRING$(24,32):V=USR(25):L=PEEK(V+1):M=PEEK(V+2):E=L+256*M:IFE)
35 DATA 327,67 THEN E=E-65536:FORI=ETOE+23:REAOC:POKEI,X:NEXTI:IFPEEK(16396)=0THE
36 DATA NPOKE16524,L:POKE16527,M:ELSE DEFUSR0=E
37 OATA 205,127,10,62,1,14,0,69,47,230,3,211,255,13,40,4,16,247,24,243,3
38 DATA 7,32,242,201
39 DATA 22, FORA=1:OT81:REAOC:IX=USR(1):NEXTA
40 DATA 25, DATA 18046,18004,10334,10340,10352,25663,18004,10334,10340,10352,1798
41 DATA 3,18004,11568,11571,11568,12857,25642,18004,10334,10340,10352,2566
42 DATA 3,25604,10334,10340,10352
43 OATA 25663,18004,10334,10340,10334,11632,30804,10340,8243,8249,8255,8
44 DATA 259,8267,8259,8255,8259,8257,11587,10324,12875,8243,8249,8255,17962,1797
45 DATA 7,10324,10315,8243,8249,8255,8259,8267,8259,8255,8259,10315,11587,8276,8
46 DATA 276,8276,11583,8262,8271
47 OATA 8276,8286,10346,10352,10366,10324,2644,8276,8276,10324,126
48 DATA 100,CLS:POKE16921,0:POKE16920,0:POKE16919,0
49 DATA 178, FORZZ=1:OT000:NEXTZZ
50 CLS:OIMA(9,9,2)
51 PRINTCHR$(23)
52 PRINTCHR$(23)
53 PRINTCHR$(23)
54 PRINTCHR$(23)
55 PRINTCHR$(23)
56 PRINTCHR$(23)
57 PRINTCHR$(23)
58 PRINTCHR$(23)
59 PRINTCHR$(23)
60 PRINTCHR$(23)
61 PRINTCHR$(23)
62 PRINTCHR$(23)
63 PRINTCHR$(23)
64 PRINTCHR$(23)
65 PRINTCHR$(23)
66 PRINTCHR$(23)
67 PRINTCHR$(23)
68 PRINTCHR$(23)
69 PRINTCHR$(23)
70 PRINTCHR$(23)
71 PRINTCHR$(23)
72 PRINTCHR$(23)
73 PRINTCHR$(23)
74 PRINTCHR$(23)
75 PRINTCHR$(23)
76 PRINTCHR$(23)
77 PRINTCHR$(23)
78 PRINTCHR$(23)
79 PRINTCHR$(23)
80 PRINTCHR$(23)
81 PRINTCHR$(23)
82 PRINTCHR$(23)
83 PRINTCHR$(23)
84 PRINTCHR$(23)
85 PRINTCHR$(23)
86 PRINTCHR$(23)
87 PRINTCHR$(23)
88 PRINTCHR$(23)
89 PRINTCHR$(23)
90 PRINTCHR$(23)
91 PRINTCHR$(23)
92 PRINTCHR$(23)
93 PRINTCHR$(23)
94 PRINTCHR$(23)
95 PRINTCHR$(23)
96 PRINTCHR$(23)
97 PRINTCHR$(23)
98 PRINTCHR$(23)
99 PRINTCHR$(23)
100 PRINTCHR$(23)
101 PRINTCHR$(23)
102 PRINTCHR$(23)
103 PRINTCHR$(23)
104 PRINTCHR$(23)
105 PRINTCHR$(23)
106 PRINTCHR$(23)
107 PRINTCHR$(23)
108 PRINTCHR$(23)
109 PRINTCHR$(23)
110 PRINTCHR$(23)
111 PRINTCHR$(23)
112 PRINTCHR$(23)
113 PRINTCHR$(23)
114 PRINTCHR$(23)
115 PRINTCHR$(23)
116 PRINTCHR$(23)
117 PRINTCHR$(23)
118 PRINTCHR$(23)
119 PRINTCHR$(23)
120 PRINTCHR$(23)
121 PRINTCHR$(23)
122 PRINTCHR$(23)
123 PRINTCHR$(23)
124 PRINTCHR$(23)
125 PRINTCHR$(23)
126 PRINTCHR$(23)
127 PRINTCHR$(23)
128 PRINTCHR$(23)
129 PRINTCHR$(23)
130 PRINTCHR$(23)
131 PRINTCHR$(23)
132 PRINTCHR$(23)
133 PRINTCHR$(23)
134 PRINTCHR$(23)
135 PRINTCHR$(23)
136 PRINTCHR$(23)
137 PRINTCHR$(23)
138 PRINTCHR$(23)
139 PRINTCHR$(23)
140 PRINTCHR$(23)
141 PRINTCHR$(23)
142 PRINTCHR$(23)
143 PRINTCHR$(23)
144 PRINTCHR$(23)
145 PRINTCHR$(23)
146 PRINTCHR$(23)
147 PRINTCHR$(23)
148 PRINTCHR$(23)
149 PRINTCHR$(23)
150 PRINTCHR$(23)
151 PRINTCHR$(23)
152 PRINTCHR$(23)
153 PRINTCHR$(23)
154 PRINTCHR$(23)
155 PRINTCHR$(23)
156 PRINTCHR$(23)
157 PRINTCHR$(23)
158 PRINTCHR$(23)
159 PRINTCHR$(23)
160 PRINTCHR$(23)
161 PRINTCHR$(23)
162 PRINTCHR$(23)
163 PRINTCHR$(23)
164 PRINTCHR$(23)
165 PRINTCHR$(23)
166 PRINTCHR$(23)
167 PRINTCHR$(23)
168 PRINTCHR$(23)
169 PRINTCHR$(23)
170 PRINTCHR$(23)
171 PRINTCHR$(23)
172 PRINTCHR$(23)
173 PRINTCHR$(23)
174 PRINTCHR$(23)
175 PRINTCHR$(23)
176 PRINTCHR$(23)
177 PRINTCHR$(23)
178 PRINTCHR$(23)
179 PRINTCHR$(23)
180 PRINTCHR$(23)
181 PRINTCHR$(23)
182 PRINTCHR$(23)
183 PRINTCHR$(23)
184 PRINTCHR$(23)
185 PRINTCHR$(23)
186 PRINTCHR$(23)
187 PRINTCHR$(23)
188 PRINTCHR$(23)
189 PRINTCHR$(23)
190 PRINTCHR$(23)
191 PRINTCHR$(23)
192 PRINTCHR$(23)
193 PRINTCHR$(23)
194 PRINTCHR$(23)
195 PRINTCHR$(23)
196 PRINTCHR$(23)
197 PRINTCHR$(23)
198 PRINTCHR$(23)
199 PRINTCHR$(23)
200 PRINTCHR$(23)
201 PRINTCHR$(23)
202 PRINTCHR$(23)
203 PRINTCHR$(23)
204 PRINTCHR$(23)
205 PRINTCHR$(23)
206 PRINTCHR$(23)
207 PRINTCHR$(23)
208 PRINTCHR$(23)
209 PRINTCHR$(23)
210 PRINTCHR$(23)
211 PRINTCHR$(23)
212 PRINTCHR$(23)
213 PRINTCHR$(23)
214 PRINTCHR$(23)
215 PRINTCHR$(23)
216 PRINTCHR$(23)
217 PRINTCHR$(23)
218 PRINTCHR$(23)
219 PRINTCHR$(23)
220 PRINTCHR$(23)
221 PRINTCHR$(23)
222 PRINTCHR$(23)
223 PRINTCHR$(23)
224 PRINTCHR$(23)
225 PRINTCHR$(23)
226 PRINTCHR$(23)
227 PRINTCHR$(23)
228 PRINTCHR$(23)
229 PRINTCHR$(23)
230 PRINTCHR$(23)
231 PRINTCHR$(23)
232 PRINTCHR$(23)
233 PRINTCHR$(23)
234 PRINTCHR$(23)
235 PRINTCHR$(23)
236 PRINTCHR$(23)
237 PRINTCHR$(23)
238 PRINTCHR$(23)
239 PRINTCHR$(23)
240 PRINTCHR$(23)
241 PRINTCHR$(23)
242 PRINTCHR$(23)
243 PRINTCHR$(23)
244 PRINTCHR$(23)
245 PRINTCHR$(23)
246 PRINTCHR$(23)
247 PRINTCHR$(23)
248 PRINTCHR$(23)
249 PRINTCHR$(23)
250 PRINTCHR$(23)
251 PRINTCHR$(23)
252 PRINTCHR$(23)
253 PRINTCHR$(23)
254 PRINTCHR$(23)
255 PRINTCHR$(23)
256 PRINTCHR$(23)
257 PRINTCHR$(23)
258 PRINTCHR$(23)
259 PRINTCHR$(23)
260 PRINTCHR$(23)
261 PRINTCHR$(23)
262 PRINTCHR$(23)
263 PRINTCHR$(23)
264 PRINTCHR$(23)
265 PRINTCHR$(23)
266 PRINTCHR$(23)
267 PRINTCHR$(23)
268 PRINTCHR$(23)
269 PRINTCHR$(23)
270 PRINTCHR$(23)
271 PRINTCHR$(23)
272 PRINTCHR$(23)
273 PRINTCHR$(23)
274 PRINTCHR$(23)
275 PRINTCHR$(23)
276 PRINTCHR$(23)
277 PRINTCHR$(23)
278 PRINTCHR$(23)
279 PRINTCHR$(23)
280 PRINTCHR$(23)
281 PRINTCHR$(23)
282 PRINTCHR$(23)
283 PRINTCHR$(23)
284 PRINTCHR$(23)
285 PRINTCHR$(23)
286 PRINTCHR$(23)
287 PRINTCHR$(23)
288 PRINTCHR$(23)
289 PRINTCHR$(23)
290 PRINTCHR$(23)
291 PRINTCHR$(23)
292 PRINTCHR$(23)
293 PRINTCHR$(23)
294 PRINTCHR$(23)
295 PRINTCHR$(23)
296 PRINTCHR$(23)
297 PRINTCHR$(23)
298 PRINTCHR$(23)
299 PRINTCHR$(23)
300 PRINTCHR$(23)
301 PRINTCHR$(23)
302 PRINTCHR$(23)
303 PRINTCHR$(23)
304 PRINTCHR$(23)
305 PRINTCHR$(23)
306 PRINTCHR$(23)
307 PRINTCHR$(23)
308 PRINTCHR$(23)
309 PRINTCHR$(23)
310 PRINTCHR$(23)
311 PRINTCHR$(23)
312 PRINTCHR$(23)
313 PRINTCHR$(23)
314 PRINTCHR$(23)
315 PRINTCHR$(23)
316 PRINTCHR$(23)
317 PRINTCHR$(23)
318 PRINTCHR$(23)
319 PRINTCHR$(23)
320 PRINTCHR$(23)
321 PRINTCHR$(23)
322 PRINTCHR$(23)
323 PRINTCHR$(23)
324 PRINTCHR$(23)
325 PRINTCHR$(23)
326 PRINTCHR$(23)
327 PRINTCHR$(23)
328 PRINTCHR$(23)
329 PRINTCHR$(23)
330 PRINTCHR$(23)
331 PRINTCHR$(23)
332 PRINTCHR$(23)
333 PRINTCHR$(23)
334 PRINTCHR$(23)
335 PRINTCHR$(23)
336 PRINTCHR$(23)
337 PRINTCHR$(23)
338 PRINTCHR$(23)
339 PRINTCHR$(23)
340 PRINTCHR$(23)
341 PRINTCHR$(23)
342 PRINTCHR$(23)
343 PRINTCHR$(23)
344 PRINTCHR$(23)
345 PRINTCHR$(23)
346 PRINTCHR$(23)
347 PRINTCHR$(23)
348 PRINTCHR$(23)
349 PRINTCHR$(23)
350 PRINTCHR$(23)
351 PRINTCHR$(23)
352 PRINTCHR$(23)
353 PRINTCHR$(23)
354 PRINTCHR$(23)
355 PRINTCHR$(23)
356 PRINTCHR$(23)
357 PRINTCHR$(23)
358 PRINTCHR$(23)
359 PRINTCHR$(23)
360 PRINTCHR$(23)
361 PRINTCHR$(23)
362 PRINTCHR$(23)
363 PRINTCHR$(23)
364 PRINTCHR$(23)
365 PRINTCHR$(23)
366 PRINTCHR$(23)
367 PRINTCHR$(23)
368 PRINTCHR$(23)
369 PRINTCHR$(23)
370 PRINTCHR$(23)
371 PRINTCHR$(23)
372 PRINTCHR$(23)
373 PRINTCHR$(23)
374 PRINTCHR$(23)
375 PRINTCHR$(23)
376 PRINTCHR$(23)
377 PRINTCHR$(23)
378 PRINTCHR$(23)
379 PRINTCHR$(23)
380 PRINTCHR$(23)
381 PRINTCHR$(23)
382 PRINTCHR$(23)
383 PRINTCHR$(23)
384 PRINTCHR$(23)
385 PRINTCHR$(23)
386 PRINTCHR$(23)
387 PRINTCHR$(23)
388 PRINTCHR$(23)
389 PRINTCHR$(23)
390 PRINTCHR$(23)
391 PRINTCHR$(23)
392 PRINTCHR$(23)
393 PRINTCHR$(23)
394 PRINTCHR$(23)
395 PRINTCHR$(23)
396 PRINTCHR$(23)
397 PRINTCHR$(23)
398 PRINTCHR$(23)
399 PRINTCHR$(23)
400 PRINTCHR$(23)
401 PRINTCHR$(23)
402 PRINTCHR$(23)
403 PRINTCHR$(23)
404 PRINTCHR$(23)
405 PRINTCHR$(23)
406 PRINTCHR$(23)
407 PRINTCHR$(23)
408 PRINTCHR$(23)
409 PRINTCHR$(23)
410 PRINTCHR$(23)
411 PRINTCHR$(23)
412 PRINTCHR$(23)
413 PRINTCHR$(23)
414 PRINTCHR$(23)
415 PRINTCHR$(23)
416 PRINTCHR$(23)
417 PRINTCHR$(23)
418 PRINTCHR$(23)
419 PRINTCHR$(23)
420 PRINTCHR$(23)
421 PRINTCHR$(23)
422 PRINTCHR$(23)
423 PRINTCHR$(23)
424 PRINTCHR$(23)
425 PRINTCHR$(23)
426 PRINTCHR$(23)
427 PRINTCHR$(23)
428 PRINTCHR$(23)
429 PRINTCHR$(23)
430 PRINTCHR$(23)
431 PRINTCHR$(23)
432 PRINTCHR$(23)
433 PRINTCHR$(23)
434 PRINTCHR$(23)
435 PRINTCHR$(23)
436 PRINTCHR$(23)
437 PRINTCHR$(23)
438 PRINTCHR$(23)
439 PRINTCHR$(23)
440 PRINTCHR$(23)
441 PRINTCHR$(23)
442 PRINTCHR$(23)
443 PRINTCHR$(23)
444 PRINTCHR$(23)
445 PRINTCHR$(23)
446 PRINTCHR$(23)
447 PRINTCHR$(23)
448 PRINTCHR$(23)
449 PRINTCHR$(23)
450 PRINTCHR$(23)
451 PRINTCHR$(23)
452 PRINTCHR$(23)
453 PRINTCHR$(23)
454 PRINTCHR$(23)
455 PRINTCHR$(23)
456 PRINTCHR$(23)
457 PRINTCHR$(23)
458 PRINTCHR$(23)
459 PRINTCHR$(23)
460 PRINTCHR$(23)
461 PRINTCHR$(23)
462 PRINTCHR$(23)
463 PRINTCHR$(23)
464 PRINTCHR$(23)
465 PRINTCHR$(23)
466 PRINTCHR$(23)
467 PRINTCHR$(23)
468 PRINTCHR$(23)
469 PRINTCHR$(23)
470 PRINTCHR$(23)
471 PRINTCHR$(23)
472 PRINTCHR$(23)
473 PRINTCHR$(23)
474 PRINTCHR$(23)
475 PRINTCHR$(23)
476 PRINTCHR$(23)
477 PRINTCHR$(23)
478 PRINTCHR$(23)
479 PRINTCHR$(23)
480 PRINTCHR$(23)
481 PRINTCHR$(23)
482 PRINTCHR$(23)
483 PRINTCHR$(23)
484 PRINTCHR$(23)
485 PRINTCHR$(23)
486 PRINTCHR$(23)
487 PRINTCHR$(23)
488 PRINTCHR$(23)
489 PRINTCHR$(23)
490 PRINTCHR$(23)
491 PRINTCHR$(23)
492 PRINTCHR$(23)
493 PRINTCHR$(23)
494 PRINTCHR$(23)
495 PRINTCHR$(23)
496 PRINTCHR$(23)
497 PRINTCHR$(23)
498 PRINTCHR$(23)
499 PRINTCHR$(23)
500 PRINTCHR$(23)
501 PRINTCHR$(23)
502 PRINTCHR$(23)
503 PRINTCHR$(23)
504 PRINTCHR$(23)
505 PRINTCHR$(23)
506 PRINTCHR$(23)
507 PRINTCHR$(23)
508 PRINTCHR$(23)
509 PRINTCHR$(23)
510 PRINTCHR$(23)
511 PRINTCHR$(23)
512 PRINTCHR$(23)
513 PRINTCHR$(23)
514 PRINTCHR$(23)
515 PRINTCHR$(23)
516 PRINTCHR$(23)
517 PRINTCHR$(23)
518 PRINTCHR$(23)
519 PRINTCHR$(23)
520 PRINTCHR$(23)
521 PRINTCHR$(23)
522 PRINTCHR$(23)
523 PRINTCHR$(23)
524 PRINTCHR$(23)
525 PRINTCHR$(23)
526 PRINTCHR$(23)
527 PRINTCHR$(23)
528 PRINTCHR$(23)
529 PRINTCHR$(23)
530 PRINTCHR$(23)
531 PRINTCHR$(23)
532 PRINTCHR$(23)
533 PRINTCHR$(23)
534 PRINTCHR$(23)
535 PRINTCHR$(23)
536 PRINTCHR$(23)
537 PRINTCHR$(23)
538 PRINTCHR$(23)
539 PRINTCHR$(23)
540 PRINTCHR$(23)
541 PRINTCHR$(23)
542 PRINTCHR$(23)
543 PRINTCHR$(23)
544 PRINTCHR$(23)
545 PRINTCHR$(23)
546 PRINTCHR$(23)
547 PRINTCHR$(23)
548 PRINTCHR$(23)
549 PRINTCHR$(23)
550 PRINTCHR$(23)
551 PRINTCHR$(23)
552 PRINTCHR$(23)
553 PRINTCHR$(23)
554 PRINTCHR$(23)
555 PRINTCHR$(23)
556 PRINTCHR$(23)
557 PRINTCHR$(23)
558 PRINTCHR$(23)
559 PRINTCHR$(23)
560 PRINTCHR$(23)
561 PRINTCHR$(23)
562 PRINTCHR$(23)
563 PRINTCHR$(23)
564 PRINTCHR$(23)
565 PRINTCHR$(23)
566 PRINTCHR$(23)
567 PRINTCHR$(23)
568 PRINTCHR$(23)
569 PRINTCHR$(23)
570 PRINTCHR$(23)
571 PRINTCHR$(23)
572 PRINTCHR$(23)
573 PRINTCHR$(23)
574 PRINTCHR$(23)
575 PRINTCHR$(23)
576 PRINTCHR$(23)
577 PRINTCHR$(23)
578 PRINTCHR$(23)
579 PRINTCHR$(23)
580 PRINTCHR$(23)
581 PRINTCHR$(23)
582 PRINTCHR$(23)
583 PRINTCHR$(23)
584 PRINTCHR$(23)
585 PRINTCHR$(23)
586 PRINTCHR$(23)
587 PRINTCHR$(23)
588 PRINTCHR$(23)
589 PRINTCHR$(23)
590 PRINTCHR$(23)
591 PRINTCHR$(23)
592 PRINTCHR$(23)
593 PRINTCHR$(23)
594 PRINTCHR$(23)
595 PRINTCHR$(23)
596 PRINTCHR$(23)
597 PRINTCHR$(23)
598 PRINTCHR$(23)
599 PRINTCHR$(23)
600 PRINTCHR$(23)
601 PRINTCHR$(23)
602 PRINTCHR$(23)
603 PRINTCHR$(23)
604 PRINTCHR$(23)
605 PRINTCHR$(23)
606 PRINTCHR$(23)
607 PRINTCHR$(23)
608 PRINTCHR$(23)
609 PRINTCHR$(23)
610 PRINTCHR$(23)
611 PRINTCHR$(23)
612 PRINTCHR$(23)
613 PRINTCHR$(23)
614 PRINTCHR$(23)
615 PRINTCHR$(23)
616 PRINTCHR$(23)
617 PRINTCHR$(23)
618 PRINTCHR$(23)
619 PRINTCHR$(23)
620 PRINTCHR$(23)
621 PRINTCHR$(23)
622 PRINTCHR$(23)
623 PRINTCHR$(23)
624 PRINTCHR$(23)
625 PRINTCHR$(23)
626 PRINTCHR$(23)
627 PRINTCHR$(23)
628 PRINTCHR$(23)
629 PRINTCHR$(23)
630 PRINTCHR$(23)
631 PRINTCHR$(23)
632 PRINTCHR$(23)
633 PRINTCHR$(23)
634 PRINTCHR$(23)
635 PRINTCHR$(23)
636 PRINTCHR$(23)
637 PRINTCHR$(23)
638 PRINTCHR$(23)
639 PRINTCHR$(23)
640 PRINTCHR$(23)
641 PRINTCHR$(23)
642 PRINTCHR$(23)
643 PRINTCHR$(23)
644 PRINTCHR$(23)
645 PRINTCHR$(23)
646 PRINTCHR$(23)
647 PRINTCHR$(23)
648 PRINTCHR$(23)
649 PRINTCHR$(23)
650 PRINTCHR$(23)
651 PRINTCHR$(23)
652 PRINTCHR$(23)
653 PRINTCHR$(23)
654 PRINTCHR$(23)
655 PRINTCHR$(23)
656 PRINTCHR$(23)
657 PRINTCHR$(23)
658 PRINTCHR$(23)
659 PRINTCHR$(23)
660 PRINTCHR$(23)
661 PRINTCHR$(23)
662 PRINTCHR$(23)
663 PRINTCHR$(23)
664 PRINTCHR$(23)
665 PRINTCHR$(23)
666 PRINTCHR$(23)
667 PRINTCHR$(23)
668 PRINTCHR$(23)
669 PRINTCHR$(23)
670 PRINTCHR$(23)
671 PRINTCHR$(23)
672 PRINTCHR$(23)
673 PRINTCHR$(23)
674 PRINTCHR$(23)
675 PRINTCHR$(23)
676 PRINTCHR$(23)
677 PRINTCHR$(23)
678 PRINTCHR$(23)
679 PRINTCHR$(23)
680 PRINTCHR$(23)
681 PRINTCHR$(23)
682 PRINTCHR$(23)
683 PRINTCHR$(23)
684 PRINTCHR$(23)
685 PRINTCHR$(23)
686 PRINTCHR$(23)
687 PRINTCHR$(23)
688 PRINTCHR$(23)
689 PRINTCHR$(23)
690 PRINTCHR$(23)
691 PRINTCHR$(23)
692 PRINTCHR$(23)
693 PRINTCHR$(23)
694 PRINTCHR$(23)
695 PRINTCHR$(23)
696 PRINTCHR$(23)
697 PRINTCHR$(23)
698 PRINTCHR$(23)
699 PRINTCHR$(23)
700 PRINTCHR$(23)
701 PRINTCHR$(23)
702 PRINTCHR$(23)
703 PRINTCHR$(23)
704 PRINTCHR$(23)
705 PRINTCHR$(23)
706 PRINTCHR$(23)
707 PRINTCHR$(23)
708 PRINTCHR$(23)
709 PRINTCHR$(23)
710 PRINTCHR$(23)
711 PRINTCHR$(23)
712 PRINTCHR$(23)
713 PRINTCHR$(23)
714 PRINTCHR$(23)
715 PRINTCHR$(23)
716 PRINTCHR$(23)
717 PRINTCHR$(23)
718 PRINTCHR$(23)
719 PRINTCHR$(23)
720 PRINTCHR$(23)
721 PRINTCHR$(23)
722 PRINTCHR$(23)
723 PRINTCHR$(23)
724 PRINTCHR$(23)
725 PRINTCHR$(23)
726 PRINTCHR$(23)
727 PRINTCHR$(23)
728 PRINTCHR$(23)
729 PRINTCHR$(23)
730 PRINTCHR$(23)
731 PRINTCHR$(23)
732 PRINTCHR$(23)
733 PRINTCHR$(23)
734 PRINTCHR$(23)
735 PRINTCHR$(23)
736 PRINTCHR$(23)
737 PRINTCHR$(23)
738 PRINTCHR$(23)
739 PRINTCHR$(23)
740 PRINTCHR$(23)
741 PRINTCHR$(23)
742 PRINTCHR$(23)
743 PRINTCHR$(23)
744 PRINTCHR$(23)
745 PRINTCHR$(23)
746 PRINTCHR$(23)
747 PRINTCHR$(23)
748 PRINTCHR$(23)
749 PRINTCHR$(23)
750 PRINTCHR$(23)
751 PRINTCHR$(23)
752 PRINTCHR$(23)
753 PRINTCHR$(23)
754 PRINTCHR$(23)
755 PRINTCHR$(23)
756 PRINTCHR$(23)
757 PRINTCHR$(23)
758 PRINTCHR$(23)
759 PRINTCHR$(23)
760 PRINTCHR$(23)
761 PRINTCHR$(23)
762 PRINTCHR$(23)
763 PRINTCHR$(23)
764 PRINTCHR$(23)
765 PRINTCHR$(23)
766 PRINTCHR$(23)
767 PRINTCHR$(23)
768 PRINTCHR$(23)
769 PRINTCHR$(23)
770 PRINTCHR$(23)
771 PRINTCHR$(23)
772 PRINTCHR$(23)
773 PRINTCHR$(23)
774 PRINTCHR$(23)
775 PRINTCHR$(23)
776 PRINTCHR$(23)
777 PRINTCHR$(23)
778 PRINTCHR$(23)
779 PRINTCHR$(23)
780 PRINTCHR$(23)
781 PRINTCHR$(23)
782 PRINTCHR$(23)
783 PRINTCHR$(23)
784 PRINTCHR$(23)
785 PRINTCHR$(23)
786 PRINTCHR$(23)
787 PRINTCHR$(23)
788 PRINTCHR$(23)
789 PRINTCHR$(23)
790 PRINTCHR$(23)
791 PRINTCHR$(23)
792 PRINTCHR$(23)
793 PRINTCHR$(23)
794 PRINTCHR$(23)
795 PRINTCHR$(23)
796 PRINTCHR$(23)
797 PRINTCHR$(23)
798 PRINTCHR$(23)
799 PRINTCHR$(23)
800 PRINTCHR$(23)
801 PRINTCHR$(23)
802 PRINTCHR$(23)
803 PRINTCHR$(23)
804 PRINTCHR$(23)
805 PRINTCHR$(23)
806 PRINTCHR$(23)
807 PRINTCHR$(23)
808 PRINTCHR$(23)
809 PRINTCHR$(23)
810 PRINTCHR$(23)
811 PRINTCHR$(23)
812 PRINTCHR$(23)
813 PRINTCHR$(23)
814 PRINTCHR$(23)
815 PRINTCHR$(23)
816 PRINTCHR$(23)
817 PRINTCHR$(23)
818 PRINTCHR$(23)
819 PRINTCHR$(23)
820 PRINTCHR$(23)
821 PRINTCHR$(23)
822 PRINTCHR$(23)
823 PRINTCHR$(23)
824 PRINTCHR$(23)
825 PRINTCHR$(23)
826 PRINTCHR$(23)
827 PRINTCHR$(23)
828 PRINTCHR$(23)
829 PRINTCHR$(23)
830 PRINTCHR$(23)
831 PRINTCHR$(23)
832 PRINTCHR$(23)
833 PRINTCHR$(23)
834 PRINTCHR$(23)
835 PRINTCHR$(23)
836 PRINTCHR$(23)
837 PRINTCHR$(23)
838 PRINTCHR$(23)
839 PRINTCHR$(23)
840 PRINTCHR$(23)
841 PRINTCHR$(23)
842 PRINTCHR$(23)
843 PRINTCHR$(23)
844 PRINTCHR$(23)
845 PRINTCHR$(23)
846 PRINTCHR$(23)
847 PRINTCHR$(23)
848 PRINTCHR$(23)
849 PRINTCHR$(23)
850 PRINTCHR$(23)
851 PRINTCHR$(23)
852 PRINTCHR$(23)
853 PRINTCHR$(23)
854 PRINTCHR$(23)
855 PRINTCHR$(23)
856 PRINTCHR$(23)
857 PRINTCHR$(23)
858 PRINTCHR$(23)
859 PRINTCHR$(23)
860 PRINTCHR$(23)
861 PRINTCHR$(23)
862 PRINTCHR$(23)
863 PRINTCHR$(23)
864 PRINTCHR$(23)
865 PRINTCHR$(23)
866 PRINTCHR$(23)
867 PRINTCHR$(23)
868 PRINTCHR$(23)
869 PRINTCHR$(23)
870 PRINTCHR$(23)
871 PRINTCHR$(23)
872 PRINTCHR$(23)
873 PRINTCHR$(23)
874 PRINTCHR$(23)
875 PRINTCHR$(23)
876 PRINTCHR$(23)
877 PRINTCHR$(23)
878 PRINTCHR$(23)
879 PRINTCHR$(23)
880 PRINTCHR$(23)
881 PRINTCHR$(23)
882 PRINTCHR$(23)
883 PRINTCHR$(23)
884 PRINTCHR$(23)
885 PRINTCHR$(23)
886 PRINTCHR$(23)
887 PRINTCHR$(23)
888 PRINTCHR$(23)
889 PRINTCHR$(23)
890 PRINTCHR$(23)
891 PRINTCHR$(23)
892 PRINTCHR$(23)
893 PRINTCHR$(23)
894 PRINTCHR$(
```



[illegible]

20 Anos  
de  
Sucesso



34





Mario Folli

Escrito para microcomputadores da linha Apple, *Órbita* é um jogo em que você deverá rastrear e destruir os satélites espíões inimigos que orbitam uma nova estrela conquistada pelo homem do século XXI.

Você possui quatro naves radares, que posicionam a mira sobre o alvo, a fim de dispararem seus raios destruidores.

Não há limite de tiro, podendo-se disparar diversas vezes sobre o mesmo satélite, mas este não é um bom procedimento, pois um ponto será decrescido do seu placar a cada disparo.

A cada satélite atingido, você receberá um bônus de dez pontos, porém perderá três, caso o satélite escape de sua perseguição.

O jogo se encerra quando vinte satélites conseguirem *furar* o seu esquema de defesa.

## O PROGRAMA

A velocidade de processamento é razoável, se considerarmos que se trata de um jogo com movimentação de figuras e escrito em BASIC. No entanto, será particularmente interessante compilá-lo

quando você já estiver craque e desejar que lhe sejam exigidas ações mais rápidas, aliadas a uma boa dose de reflexos e coordenação de teclas.

Na compilação, é recomendável que se altere o loop vazio FOR/NEXT da linha 2500, de 30 para 40 vezes, para que as velocidades dos satélites não sofram variações quando sua mira estiver parada na tela.

## PARTES DO PROGRAMA

- 50-98 : Instruções do jogo
- 210-300 : Definição das velocidades dos satélites, posição onde surgirão no vídeo e os tipos de satélites (total de seis).
- 350-380 : Finalização de uma partida
- 400-440 : Rotina que altera a estrela no centro do vídeo
- 500-650 : Movimentação dos satélites na horizontal
- 700-850 : Movimentação dos satélites na vertical
- 1000-1160 : Formatação das figuras utilizadas, a partir do endereço 24576 (decimal)

apple apple apple appl

# Órbita

= início da página 2 de alta resolução

1500-1580 : Rotina que desenha a tela de jogo

2000-2500 : Rotina de posicionamento da mira

## VARIÁVEIS UTILIZADAS

- V% : Velocidade das naves satélites
- L% : Lado da tela em que os satélites surgirão
- N% : Tipo de nave satélite que irá surgir
- S% : Contador de naves satélites
- P% : Código das teclas pressionadas
- PT% : Pontos marcados
- RE% : Placar recorde
- NL% : Naves satélites não atingidas
- XI% : Posição horizontal inicial
- XF% : Posição horizontal final
- YI% : Posição vertical inicial
- YF% : Posição vertical final

Mário Folli é Engenheiro Civil, formado pela Escola de Engenharia Mauá. Atualmente, trabalha como Analista da Área de Repasses do Banco Itaú, onde desenvolve sistemas em microcomputadores Itautech. Ele também é Instrutor de linguagem BASIC, na S.O.S. Computadores, e usuário de um Apple II.

```

50 REM APPLE II * ORBITA * OUT/
85 * MARIO FOLLI *
60 HGR : TEXT : HOME : INVERSE
70 PRINT : PRINT SPC(14); "O R
  B I T A"; SPC(15); NORMAL
80 PRINT : PRINT "SEU OBJETIVO N
  ESTE JOGO, E' A DESTRUIÇÃO D
  SATÉLITES ESPÍOES QUE ORBIT
  AM UMA NO-VA ESTRELA CONQUI
  TADA PELO HOMEM."
82 PRINT "PARA CUMPRIR ESTA MISS
  AO, POSICIONE A MI-RA E DISPA
  RE RAIOS CONTRA OS ESTRANHOS
  ."
84 PRINT "O JOGO TERMINA QUANDO
  20 SATÉLITES ESCA-PAREM DE S
  UA PERSEGUIÇÃO." : GOSUB 1000

86 VTAB 12: HTAB 1: INVERSE : PRINT
  SPC(11); "TECLAS UTILIZADAS
  ~: SPC(12); NORMAL : PRINT
  SPC(40);

88 PRINT "ACIMA Z:BAIXO (-:ES
  Q. -):DIR. BEI-RO": SPC(40
  ); INVERSE
90 PRINT SPC(11); "P O N T U A
  C A O": SPC(12); NORMAL
92 PRINT : PRINT SPC(3); "SATELI
  TE ATINGIDO.....+10 PONTOS"

94 PRINT SPC(3); "SATELITE ESCAP
  A.....-03 PONTOS"
96 PRINT SPC(3); "TIRO DISPARADO
  .....-01 PONTOS"

98 GOSUB 1500: INVERSE : PRINT :
  PRINT "PRESSIONE QUALQUER
  TECLA PARA O INICIO ~: VTAB
  21: HTAB 20: GET A$: NORMAL :
  HOME

100 POKE - 16304,0: POKE - 162
  97,0: POKE - 16300,0: REM
  RECUPERA PAG.1 ALTA RES.
200 VTAB 22: HTAB 5: PRINT "PONT
  OS": PTX = 0: HTAB 30: PRINT
  "ORBITA": HTAB 5: PRINT "RE
  CORD": REZ = 0: HTAB 30: PRINT
  "NAVES": NLX = 0: GOSUB 400

210 VX = RND(1) * 6: IF VX < 2 THEN
  210
220 LX = RND(1) * 5: IF LX = 0 THEN
  220
230 NX = RND(1) * 7: IF NX = 0 THEN
  230
240 SX = SX + 1: VX = VX + (SX) * 1
  5) + (SX) * 30
250 IF NLX = 0 THEN 350
300 ON LX GOTO 500,500,700,700
350 FLASH : VTAB 21: HTAB 5: PRINT
  "FIM DE JOGO : OUTRA VEZ $
  /N)?": GET A$: IF A$ = "N" THEN
  TEXT : HOME : NORMAL : END

360 IF A$ ( ) "S" THEN PRINT CHR$
  (7): GOTO 350
370 IF PTX : REZ THEN REZ = PTX
380 NLX = 0: PTX = 0: HOME : NORMAL
  : GOTO 200
400 REM MODIFICA A ESTRELA
410 FOR I = 1 TO 3: HX = RND(1)
  * 7: SX = RND(1) * 15: IF
  SX = 0 THEN SX = 15
  HCOLOR = HX: SCALE = SX: FOR J
  = 0 TO 64 STEP 4: ROT = J: DRAW
  7 AT 140,80: NEXT
430 POKE 0,5 + INT( RND(1) *
  100): POKE 1,5 + INT( RND
  (1) * 30): CALL 771: NEXT
  SCALE = 1: HCOLOR = 3: ROT = 0:
  RETURN
500 REM ESQUERDA (-) DIREITA
510 YNX = RND(1) * 140: IF YNX (
  21 THEN 510
520 XIX = 50: XFZ = 230
530 IF LX = 2 THEN VX = - VX: XI
  Z = 230: XFZ = 50
540 FOR M = XIX TO XFZ STEP VX
  XDRAW NX AT M, YNX: PX = PEEK
  ( - 16384): IF PX = 160 THEN
  610
600 GOSUB 2100
  XDRAW NX AT M, YNX: NEXT : NLX
  = NLX + 1: PTX = PTX - 3: GOTO
  200
610 POKE 0,200: POKE 1,8: CALL 7
  71: SCALE = 3: XDRAW 1 AT XX,
  YZ: POKE 0,100: POKE 1,8: CALL
  771: XDRAW 1 AT XX, YZ: SCALE =
  1: PTX = PTX - 1
620 IF M > XX - 5 AND M < XX + 5
  AND YNX > YZ - 5 AND YNX <
  YZ + 5 THEN 650
630 POKE - 16368,0: GOTO 600
650 PTX = PTX + 10: FOR I = 1 TO
  5: SCALE = I: XDRAW NX AT XX,
  YZ: POKE 0,100: POKE 1,8: CALL
  771: XDRAW 1 AT XX, YZ: SCALE =
  1: PTX = PTX - 1
680 GOSUB 2100
  XDRAW NX AT XNX, M: NEXT : NLX
  = NLX + 1: PTX = PTX - 3: GOTO
  200
810 POKE 0,200: POKE 1,8: CALL 7
  71: SCALE = 3: XDRAW 1 AT XX,
  YZ: POKE 0,100: POKE 1,8: CALL
  771: XDRAW 1 AT XX, YZ: SCALE =
  1: PTX = PTX - 1
820 IF M > YZ - 5 AND M < YZ + 5
  AND XNX > XX - 5 AND XNX <
  XX + 5 THEN 850
830 POKE - 16368,0: GOTO 800
850 PTX = PTX + 10: FOR I = 1 TO
  5: SCALE = I: XDRAW NX AT XX,
  YZ: POKE 0,100: POKE 1,8: CALL
  771: XDRAW NX AT XX, YZ:
  : NEXT : SCALE = 1: POKE - 1
  6368,0: XDRAW NX AT XNX, M: GOTO
  200
1000 REM FORMATAÇÃO DAS FIGURAS
1010 DATA 173,48,192,136,208,4,
  198,1,240,8,202,208,246,166,
  0,76,3,3,96
1020 DATA 10,0,22,0,44,0,74,0,1
  06,0,136,0,172,0,203,0,209,0,
  230,0,251,0
1030 DATA 9,45,61,63,60,60,36,5
  2,54,55,55,63,47,45,46,46,54,
  38,36,37,37,0
1040 DATA 33,36,60,44,53,55,54,
  63,63,55,39,44,46,45,54,54,6
  2,46,37,39,36,45,45,37,53,62
  ,60,63,4,0
1050 DATA 42,53,54,45,61,63,36,
  36,36,36,45,61,63,54,62,63,3
  9,36,63,47,45,54,54,54,63,
  47,45,36,44,5,0
1060 DATA 36,36,45,60,63,55,45,
  54,54,55,55,63,36,55,54,46,3
  6,45,44,44,53,53,45,36,53,54
  ,62,36,7,0
1070 DATA 45,45,36,53,54,62,36,
  63,63,36,36,63,44,45,53,63,5
  4,54,63,63,54,39,36,44,54,45
  ,45,54,54,63,46,45,37,63,4,0
1080 DATA 45,45,53,54,39,36,63,
  63,63,63,39,36,53,54,45,45,3
  6,36,44,45,62,63,54,54,54,54
  ,62,63,44,45,0
1090 DATA 33,63,54,45,4,0
1100 DATA 38,45,53,9,52,27,36,3
  9,39,39,52,62,62,62,54,27,52
  ,9,44,45,0
1110 DATA 52,63,39,27,38,9,54,5
  3,53,38,44,44,44,36,9,38,
  27,62,63,0
1120 DATA 52,38,47,61,13,9,36,3
  6,63,63,63,63,54,54,54,45,
  45,45,45,36,36,0
1130 FOR I = 771 TO 789: READ AX
  : POKE I, AX: NEXT : PRINT
1140 PX = 16384: FOR I = 1 TO 274
  : READ AX: POKE PX, AX: PX = P
  X + 1: PRINT ".": NEXT
1150 POKE 232,0: POKE 233,64: REM
  ENDEREÇO DAS SHAPE TABLES
  16384/256=64
1160 RETURN
1500 REM DESENHA TELA INICIAL
1510 HCOLOR = 3: HPLT 30,0 TO 25
  0,0 TO 250,159 TO 30,159 TO
  30,0 TO 45,15 TO 235,15 TO 2
  35,145 TO 45,145 TO 45,15
1520 FOR I = 1 TO 100
1530 XX = RND(1) * 235: IF XX (
  45 THEN 1530
1540 YZ = RND(1) * 145: IF YZ (
  15 THEN 1540
  HPLT XX, YZ: NEXT : ROT = 0:
  SCALE = 1
1560 HPLT 250,0 TO 235,15: HPLT
  250,160 TO 235,145: HPLT 30
  ,160 TO 45,145: HCOLOR = 6
1570 FOR Y = 15 TO 145: HPLT 1,
  Y TO 21, Y: HPLT 259, Y TO 27
  9, Y: NEXT : XX = 140: YX = 80
1580 XDRAW 8 AT XX,155: XDRAW 9 AT
  XX,5: XDRAW 10 AT XX, YZ: ROT =
  16: XDRAW 8 AT 35, YZ: XDRAW
  9 AT 245, YZ: ROT = 0: NLX = 0:
  PTX = 0: INX = 0: RETURN
2000 REM MOVIMENTOS DA MIRA
2100 IF PX = 136 AND XX > 50 THEN
  XDRAW 9 AT XX,5: XDRAW 8 AT
  XX,155: XDRAW 10 AT XX, YZ: XX
  = XX - 5: XDRAW 9 AT XX,5: XDRAW
  8 AT XX,155: XDRAW 10 AT XX,
  YZ: RETURN
2200 IF PX = 149 AND XX < 230 THEN
  XDRAW 9 AT XX,5: XDRAW 8 AT
  XX,155: XDRAW 10 AT XX, YZ: XX
  = XX + 5: XDRAW 9 AT XX,5: XDRAW
  8 AT XX,155: XDRAW 10 AT XX,
  YZ: RETURN
2300 IF PX = 193 AND YX > 20 THEN
  ROT = 16: XDRAW 8 AT 35, YZ: XDRAW
  9 AT 245, YZ: ROT = 0: XDRAW 1
  0 AT XX, YZ: YX = YX + 5: ROT =
  16: XDRAW 8 AT 35, YZ: XDRAW
  9 AT 245, YZ: ROT = 0: XDRAW 1
  0 AT XX, YZ: RETURN
2400 IF PX = 218 AND YX < 140 THEN
  ROT = 16: XDRAW 8 AT 35, YZ: XDRAW
  9 AT 245, YZ: ROT = 0: XDRAW 1
  0 AT XX, YZ: RETURN
2500 FOR F = 1 TO 30: NEXT : RETURN

```

Órbita



## PROGRAMA PARA TK 2000 CIÊNCIA MODERNA DE COMPUTAÇÃO LTDA.

Av. Rio Branco, 156 - Sub-Solo - Loja 127 - CEP 20.043 - Centro - RJ  
Tels.: (021) 262-5723 ou 240-9327



ÍNDICE DE ACÃO EM LINGUAGEM DE MÁQUINA.

ÍNDICE DE ACÃO	EM LINGUAGEM DE MÁQUINA	ENDEREÇO
017 BOOLEAN	TIPO COM-CONE	EM PITA 30.000
018 BABOTAGE	INDEFENSA-SE DOS PARABOLISTAS	EM PITA 30.000
019 PANICO	TRATE OS MOVIMENTOS NA CONSTRUÇÃO	EM PITA 30.000
020 ATABUS	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
021 BUNO BUSTLES	DESENVOLVA O LÁBIRO	EM PITA 30.000
022 INVASORES	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
023 AUTO ESTRADA	TIPO COM-CONE	EM PITA 30.000
024 HEAS	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
025 BLITZ	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
026 U.P.O.	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
027 OTHOLD	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
028 MINUSCULAS	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
029 SCHWARZ	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
030 VADREZ	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
031 PUL-BO-BAPO	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
032 FILTERANA	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
033 CEILING ZERO	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
034 BOMBOU	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
035 MAZEMAN	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
036 SUICIDIO	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
037 ELIMINATOR	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
038 HIGHT MISSION	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
039 BUD-ATTACK	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
040 ASTEROIDS	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
041 BRAN-PRIX	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
042 SPACE EGGS	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
043 GARRA-DO-LEITE	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
044 GALAXIANS	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
045 SIMULADOR DE VOO	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
046 DEFENSE	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
047 WORMHOLE	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
048 LUNAR	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
049 ESPACEMAN	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
050 DEATH STAR	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
051 ESCALADA	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
052 HORAS	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
053 CALCULADORA	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
054 MALA DIRETA	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
055 MALA DIRETA	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
056 CONTROLE DE CLIENTES	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
057 CALC-7000	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
058 CALC-7000	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
059 CONTROLE DE CLIENTES	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
060 MULTICALL	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
061 BANCO DE DADOS	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
062 LOTO	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
063 GRAPLOT	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
064 CONTROLE DE ESTOQUE	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
065 CONTAS A RECEBER	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
066 CONTAS A RECEBER	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000
067 CONTABILIDADE	INDEFENSA-SE DAS NAVES INVASORES	EM PITA 30.000

Desejo receber os seguintes programas pelo(s) qual(is) pagarei a quantia de Cr\$ \_\_\_\_\_

Nºs PROGRAMAS: \_\_\_\_\_

NOME: \_\_\_\_\_

END.: \_\_\_\_\_

CIDADE: \_\_\_\_\_ UF.: \_\_\_\_\_ CEP: \_\_\_\_\_

Para tal estou enviando um cheque nominal à Ciência Moderna de Computação Ltda. endereçado à ATI, Av. Presidente Wilson, 165 Grupo 1210 - Centro, CEP 20.030 - Rio de Janeiro-RJ. Despesas de Correio inclusa.





## Batalha Naval

Renato da Silva Oliveira e Pierluigi Piazza

O programa que apresentamos a seguir, ilustra como uma idéia simples pode dar origem a um jogo de razoável complexidade. Uma das coisas mais triviais que se pode pensar, ao fazer um jogo, é num atirador e num alvo. Obviamente, nesses casos, o usuário costuma fazer o papel do atirador. Em nosso exemplo, usamos um navio lançando cargas em profundidade num submarino.

Um pouco mais difícil é engendrar uma forma de fazer com que o usuário tenha que pensar antes de dar seu tiro. Em nosso jogo é praticamente inevitável o uso do fosfato cerebral! O usuário é obrigado a pensar para obter êxito (e isso, para muitos, dói!).

A idéia por trás de tudo é bastante simples. Para atingir o submarino, o usuário deve dar a posição em que a carga será lançada e isso é feito através da introdução de três números: um que indique a posição na direção Norte-Sul, outro que indique a posição na direção Leste-Oeste e um terceiro indicando a profundidade em que a carga deverá explodir.

O submarino é colocado, ao acaso, dentro de uma região cúbica de 8 Km na direção Norte-Sul, 8 Km na direção Leste-Oeste e 4 Km de profundidade. Nessa mesma região, está colocado também o navio lançador de cargas. Observe a figura 1.

Note que o navio sempre fica no vértice do paralelepípedo formado pela região de 8x8x4 quilômetros cúbicos e que o submarino é sempre colocado numa posição de coordenadas inteiras. Por exemplo, ele nunca vai estar em uma profundidade de 2,5 Km (estará a 2 ou 3 Km).

Ao ser executado, o programa pede a introdução dos três números indicadores

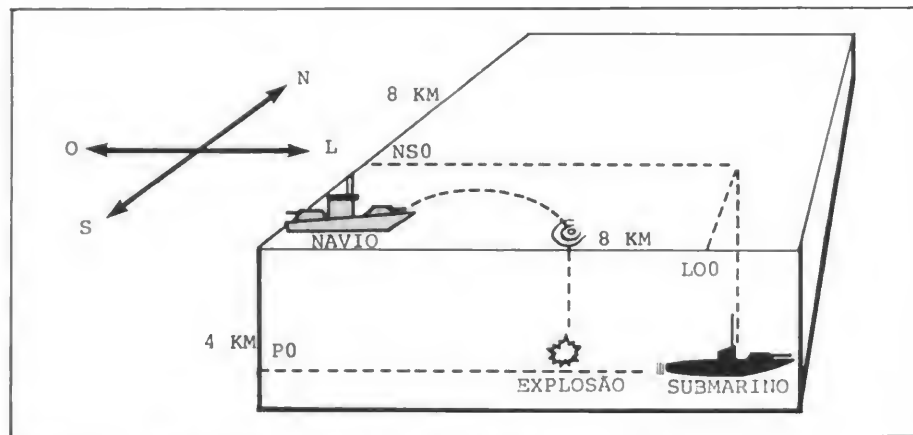


Figura 1 — O navio e o submarino estão numa região de 8x8x4 Km³.

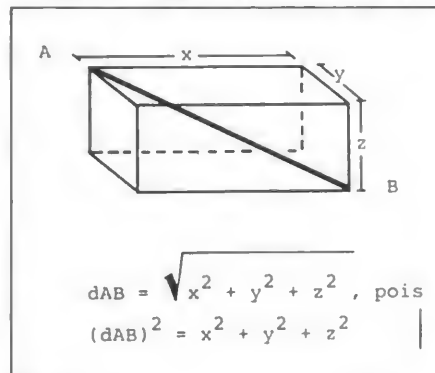


Figura 2

da posição. Inicialmente, a distância ao longo da direção Norte-Sul (N/S), depois a distância ao longo da direção Leste-Oeste (L/O) e, finalmente, a profundidade (P). O primeiro tiro deve ser dado ao acaso. Lembre-se que os valores para N/S e L/O devem estar entre 0 e 8, e os valores para P devem estar entre 0 e 4. Caso esses limites sejam desrespeitados, o tiro será perdido.

Como resposta a cada tiro, o programa dá uma dica ao usuário: a distância a que a carga explodiu do submarino. Essa

distância é calculada através do "Teorema de Pitágoras" aplicado a um espaço tridimensional (Teorema de "Tritágoras"). A ilustração do teorema pode ser vista na figura 2.

É a partir dessa informação que os tiros subsequentes devem ser dados. Por exemplo, imagine que seu primeiro tiro tenha sido 4 1 3 e que a distância obtida como dica tenha sido 4. Isso significa que a distância em que a carga lançada explodiu do submarino foi de 4 Km.

O segundo lançamento já deve ser feito no sentido de verificar a posição do submarino numa dada direção. Considerando ainda a situação exemplificada anteriormente, suponha que queremos verificar profundidade. Basta manter os número N/S e L/O e alterar — para mais ou menos — o número P. Por exemplo, imagine que fizemos o lançamento 4 1 2 e que recebemos a dica 5.196. Isso significa que nos afastamos da posição procurada e é conveniente tentarmos outro sentido.

Se tentarmos o lançamento 4 1 4

e obtivermos o mesmo número como dica (5.196), podemos então concluir que a profundidade do submarino é de 3 Km. Agora, devemos fazer o mesmo com os números N/S e L/O.

Depois de algumas partidas mal sucedidas, você achará o caminho certo para destruir o submarino. Eventualmente, após adquirir bastante prática, o leitor pode tentar alterar dois ou três números de uma só vez.

Vamos, agora, completar o programa de modo a sofisticá-lo um pouco. Antes, porém, analisemos suas partes principais:

**Inicialização** — Esta parte faz a apresentação na tela, recebe o nome de jogador através do teclado e sorteia a posição do submarino na região delimitada para o jogo. O sorteio é realizado através da função RND. Como a posição do submarino é dada através de três números, basta fazer um sorteio para cada um deles (veja as linhas 80 a 100). Se você quiser alterar a região onde o submarino pode estar, basta mudar os números 9 e 5 nas linhas 80, 90 e 100.

**Lançamentos** — Esta parte recebe os dez lançamentos do jogador. Cada lançamento necessita da introdução de três

SOM DE FUNDO,	
30	INTERVAL ON
40	ON INTERVAL=150 GOSUB 470
470	SOUND 6,8
480	SOUND 7,55
490	SOUND 10,9
500	SOUND 11,0
510	SOUND 12,180
520	SOUND 13,10
530	RETURN
SOM DO LANÇAMENTO E DA EXPLOSAO DA CARGA,	
270	GOSUB 540
540	SOUND 6,20
550	SOUND 7,55
560	SOUND 8,16
570	SOUND 11,0
580	SOUND 12,180
590	SOUND 13,10
600	RETURN
SOM DE EXPLOSAO DO NAVIO,	
330	GOSUB 540

Figura 3

números. Após receber cada um deles através do teclado, o programa calcula a distância entre a posição da explosão e a posição do submarino (através do Teorema de "Tritágoras"). Você lembra?!). Finalmente, os dados introduzidos e a distância calculada são impressos na tela no formato de uma tabela.

**Finalização** — Esta parte simplesmente encerra o jogo, informando a vitória ou derrota do jogador.

Agora sim, vamos incrementar o programa! Uma das grandes vantagens dos micros da linha MSX sobre os outros, está na facilidade com que podemos produzir sons através de seu BASIC. Acrescente ao programa as linhas que constam na figura 3. Todas elas tornam o programa um pouco mais atrativo. Agora, você pode começar a fazer suas próprias modificações acrescentando outros sons ou fazendo com que o navio e o submarino sejam desenhados na tela.

Uma mudança mais estrutural seria fazer com que, após cada tiro, o submarino se deslocasse. Isso tornaria o jogo muito mais difícil.

O programa deve ser digitado, mantendo-se a numeração das linhas da listagem. Posteriormente, você deverá completá-lo com as linhas que estão faltando.

Pierluigi Piazza é formado em Química e Física pela USP. Foi professor do curso Angelo Vestibulares e já editou a revista Micro-hobby. É autor de vários livros e atualmente dirige a Aleph Publicações, responsável por muitos sucessos na área editorial, dada a sua orientação didática.

Batalha Naval

MICRO SISTEMAS, Janeiro/86

**DADOS**  
sistemas de banco de dados

## CURSOS DISPONÍVEIS

- Introdução à Microcomputação
- DOS — PC "Sistema Operacional"
- UNIX "Sistema Operacional"
- LINGUAGEM C "Ling. Programação"
- DBASE II "Programação Básica"
- DBASE II "Program. Avançada"
- DBASE III "Program. Básica"
- LOTUS 1-2-3 "Plan. Eletrônica"
- Framework "Sistema Integrado"
- Symphony "Sistema Integrado"
- Wordstar "Processador de Texto"

\* Em desenvolvimento

**MATERIAIS DIDÁTICOS:** Publicações Técnicas desenvolvidas em português.

**RECURSOS DIDÁTICOS:** Conceitos e exemplos práticos, através de Micros e Telão de 72"

**CURSOS FECHADOS E ABERTOS**

NOVO ENDEREÇO  
Rua Graziândia, 1.750 - Jardim Europa  
CEP - 01434 - São Paulo - SP  
Telefones: 881-4599 - 881-4032

## "MIKROS"

- Microcomputadores Pessoais e Profissionais, Software, Suprimentos e Cursos.
- Financiamento em até 18 meses sem entrada e os preços mais baixos do mercado.
- Atendimento perfeito, profissionais treinados e habilitados para dar a você a certeza de um bom Investimento.

### EQUIPAMENTOS

Micros das linhas: TRS-80 • Apple • IBM • Sinclair • TRS-80 Color Impressoras • Vídeos • Interfaces • Etc.

### SUPRIMENTOS

Formulário Contínuo • Disquetes • Fitas • Mesas • Etiquetas • Etc.

### SOFTWARE

Nacionais e Importados mais de 2.000 programas e jogos de todas as linhas.

### CURSOS

Basic • Basic Avançado e DOS.

Av. Ataúfo de Paiva, 566  
sobreloja 211 e 202  
Rio de Janeiro - R.J.  
Tels.: (021) 239-2798 e 511-0599

# Olhadinha

Francisco Corrêa

Trata-se de um jogo, do tipo educativo, para todos os computadores compatíveis com o TRS-Color. Com ele, crianças numa faixa de cinco a nove anos de idade poderão memorizar, brincando, a posição das teclas, bem como incrementar sua capacidade de concentração e acuidade visual.

O funcionamento do *Olhadinha* é muito simples: após o comando <RUN>, um menu aparecerá na tela, junto com uma simulação de um semáforo, à direita do vídeo. Um *bip* indicará cada mudança de luz, do amarelo ao verde. Logo em segui-

da, o programa fará aparecer, por um breve instante, uma palavra dentro de uma espécie de janela desenhada na tela. O desafio é digitar, no espaço indicado, a mesma palavra e conseguir abiscoitar dez pontos pelo acerto.

*Francisco J. M. Corrêa da Silva é Engenheiro Eletrônico e trabalha atualmente na Xerox do Brasil, como Especialista de Marketing. Lida com computação desde 1981 e é fundador do TRS-80 Color Club.*

```

5 REM *****
6 REM * TRS-80 COLOR CLUBE RJ *
7 REM * C.POSTAL 2951 - 20001 *
8 REM * POR: FRANCISCO CORREA *
9 REM *****
10 CLS
20 PRINT@96,"A ESQUERDA DA TELA EXISTE UMA":PRINT
SINALEIRA IGUAL A DE TRANSITO."
30 PRINT"UMA PALAVRA SURGIRA POR UM INS- ANTE NA T
ELA E VOCE TERA DE DI- GITAR A RESPOSTA CORRETA."
40 PRINT"VOCE GANHA 10 PONTOS PARA CADA RESPOSTA
CORRETA. SE A RESPOSTA ESTIVER ERRADA, VOCE TERA U
MA SEGUNDA CHANCE, MAS NAO GANHARA PONTOS."
80 REM
90 PRINT@42," Olhadinha ";
100 FORX=1 TO 200:NEXTX
110 SC=RND(0)
120 PRINT@42," OLHADINHA ";
125 SOUND200,1
130 FOR X=1 TO 200:NEXTX
135 PRINT@457,"QUALQUER TECLA";
140 IF INKEY#="" THEN 90
150 F=250
160 CLS 3
170 GOSUB 770
180 FOR X=1 TO 30
190 PRINT@X,CHR$(179);
200 PRINT@X+256,CHR$(188);
210 NEXT
220 FOR Y=32 TO 256 STEP 32
230 PRINT@Y,CHR$(181);
240 PRINT@Y+31,CHR$(186);
250 NEXT
260 PRINT@73,"*OLHADINHA*";
270 FOR X=136 TO 149
280 PRINT@X,CHR$(191);
290 PRINT@X+64,CHR$(191);
300 NEXT
310 PRINT@168,CHR$(191);:PRINT@181,CHR$(191);
320 T=T+1
330 IF T>10 THEN GOSUB 870
340 Z=RND(30)
350 IF Z=N THEN 340
360 N=Z
370 FOR F=1 TO N
380 READ N$
390 NEXT
400 PRINT@289," >>>DIGITE A RESPOSTA AQUI<<<"
410 REM
420 L$=CHR$(128)
430 FOR Q=89 TO 219 STEP 32
440 PRINT@Q,L$+L$+L$;:NEXTQ
450 PRINT@418,"10 PONTOS PARA CADA ACERTO";
460 PRINT@122,CHR$(191);
465 SOUND10,1
470 GOSUB 940
480 PRINT@122,L$;
490 PRINT@154,CHR$(255);
495 SOUND10,1
500 GOSUB 940
505 SOUND10,1
510 PRINT@154,L$;
520 PRINT@186,CHR$(143);
530 GOSUB 940
540 PRINT@186,L$;
550 PRINT@169,N$;
560 FOR TL=1 TO F:NEXT
570 FOR B=169 TO 180
580 PRINT@B,CHR$(175);
590 NEXT B
600 GOSUB 780
610 PRINT@289," ":INPUT"QUAL A PALAVRA ";W$
620 IF W$<>N$ THEN GOTO700
630 IF TRY=1 OR TRY=2 AND W$=N$ THEN GOTO 670
640 IF W$=N$ THEN GOSUB 760
650 PRINT@300,"correto";
660 GOSUB 940
670 TRY=0 :GOSUB 770
680 RESTORE
690 F=F-50;GOTO320
700 PRINT@291,"a" L$ "resposta" L$ "esta" L$ "errada..."
710 F=F+100
720 IF F>250 THEN F=250
730 TRY=TRY+1
740 IF TRY=2 THEN GOTO 800
750 GOTO 420
760 CO=CO+10
770 PRINT@481,"VOCE ATE AGORA TEM: "CO "PONTOS";
780 PRINT@288," "
790 RETURN
800 PRINT@302,"*":M=LEN(N$)
810 PRINT@290," A RESPOSTA E...";
820 FOR A=1 TO M
830 PRINT@306+A,MID$(N$,A,1);
840 SOUND RND(230),2:NEXT
850 GOSUB 940
860 TRY=0: RESTORE:GOTO 320
870 PRINT@355,"JOGA DE NOVO <S> OU <N>";
880 G$=INKEY$;IF G$="" THEN 870
890 IF G$="S" THEN 910
900 IF G$="N" THEN CLS4:END
910 RESTORE:F=250;CO=0:T=0
920 CLS:GOTO150
930 GOTO 320
940 FOR TY= 1 TO 600:NEXT TY
950 RETURN
960 DATA RISCAR,ACREDITAR,RUA,MACACO,DINHEIRO,PAO
970 DATA REAL,CADEIA,FLOR,PEDRA,SELO,MEL
980 DATA HOTEL,VILA,AEROPORTO,DADOS,BATALHA,DESTE
990 DATA PLUMA,MALA,BARRO,ANEL,LAMPADA,FORCA
1000 DATA CAO,GATO,FRIO,OURO,CARA,CORDA

```

Olhadinha

Micro  
FichasTRS-COLOR  
CÓDIGOS E MENSAGEM DE ERROMicro  
Sistemas

7/0 Tentativa de divisão por zero.	IE Tentativa de ler um dado além do fim do arquivo. Use as instruções EOF e LOF para descobrir se você já atingiu o fim de um arquivo, neste caso feche-o com (CLOSE).
AO Tentativa de reabrir arquivo. Aparece também se o RESET for pressionado durante operação com gravador.	IO Erro de entrada ou saída. Ocorre frequentemente nas operações de gravação com cassete.
BS Os índices de uma matriz estão fora da faixa. Use DIM para dimensionar a matriz. Ex.: PRINT A(12), dará este erro se A não tiver sido pré-dimensionado com pelo menos 12 elementos.	LS String muito longa. Cada string pode conter no máximo 255 caracteres.
CN Tentativa de usar o comando CONT em momento inadequado. Ex.: Após o processamento da última linha do programa.	NF NEXT sem FOR. Foi utilizada uma instrução NEXT sem a respectiva instrução FOR.
DD Tentativa de redimensionar uma matriz. Cada matriz só pode ser dimensionada uma vez. Ex.: Não seria possível usar DIM A(10) e DIM A(50) no mesmo programa.	NO Tentativa de enviar dados para um arquivo ainda não aberto (OPEN).
DN Uso ilegal de número de dispositivo. Com OPEN; CLOSE; PRINT ou INPUT você só pode usar 0, -1 ou -2.	OD READ sem DATA. Os dados das instruções DATA são insuficientes para a instrução READ.
DS Instrução direta no arquivo de dados sem a especificação do número de linha. Pode ser causado se você tentar carregar um arquivo de dados em fita.	OM Falta memória. Toda a memória disponível foi usada ou reservada.
FC Chamada incorreta de função. Foi usada uma função com um valor ilegal nos seus parâmetros. Ex.: SOUND (260, 260) ou A(1).	OS Falta espaço para strings. Deve-se limpar espaços com CLEAR.
FD Formato de campo incorreto. Ocorre quando se lê (INPUT) ou grava (PRINT) um dado usando formato inadequado. Ex.: INPUT # -1,A quando o dado no arquivo é uma string.	OV Estouro de variável numérica. Foi gerado um número muito grande ou muito pequeno que o computador não tem capacidade para manipular.
FM Modo de arquivo inválido. Ocorre quando se tenta entrar com dados em um arquivo aberto apenas para saída (O) ou imprimir dados de um arquivo aberto para entrada (I).	RG RETURN sem GOSUB. Foi encontrada uma instrução RETURN, sem um GOSUB correspondente.
ID Instrução direta ilegal. Ex.: Tentar usar INPUT sem numeração de linha.	SN Erro de sintaxe. Uso incorreto de pontuação; parênteses; aspas; caráter ilegal ou comando mal digitado. Redigite a linha ou comando.
	ST Fórmula de string muito complexa. Foi usada uma operação muito complexa dentro de uma string. Divida em operações menores para funcionar.
	TM Incompatibilidade de tipo de variável. Foi tentada uma atribuição incompatível com a string utilizada. Ex.: A = "dado".
	UL Linha indefinida. Foi solicitado um desvio (GOTO; GOSUB), para uma linha inexistente.

ATI Produção

## Economize tempo e dinheiro.

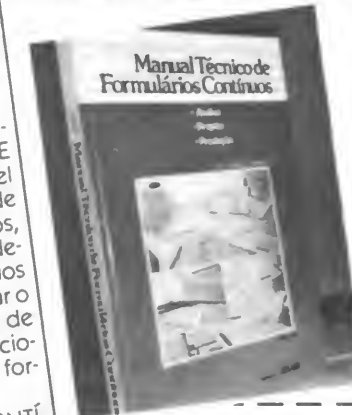
Racionalize seu 'business forms'

A EDITORA JOLAN acaba de preencher uma lacuna do mercado de publicações especializadas, com o lançamento do livro **MANUAL TÉCNICO DE FORMULÁRIOS CONTÍNUOS**, de Luiz Carlos Cardoso, um dos mais renomados profissionais do ramo.

Um verdadeiro curso de especialização em "business forms", o livro **MANUAL TÉCNICO DE FORMULÁRIOS CONTÍNUOS** é indispensável para usuários de informática, profissionais de administração, O&M, processamento de dados, analistas de sistemas, programadores, vendedores e compradores de Formulários Contínuos e técnicos de empresas. Ensina como projetar o formulário em função do seu equipamento de processamento, tendo em vista também a racionalização dos custos industriais do próprio formulário.

**MANUAL TÉCNICO DE FORMULÁRIOS CONTÍNUOS** contém 248 páginas impressas em papel off-set, com 206 ilustrações, fotos e tabelas, no formato 18x27cm, capas a 4 cores em papel especial, com acabamento costurado e encadernado.

Por se tratar de um livro de caráter técnico-didático, a aquisição de exemplares por empresas ou pessoas físicas possibilita desconto no Imposto de Renda, além de incentivos fiscais por apoio cultural e técnico.



Preço por exemplar  
Cr\$ 250.000  
Validade: até 28/2/86

Desejo receber ( ) exemplar(es) do livro Manual Técnico de Formulários Contínuos, pelo(s) qual(is) pagarei a quantia de Cr\$ \_\_\_\_\_ (\_\_\_\_\_).

Nome \_\_\_\_\_  
Endereço \_\_\_\_\_  
Cidade \_\_\_\_\_ Estado \_\_\_\_\_ CEP \_\_\_\_\_

Para tal estou enviando um cheque nominal à Editora JOLAN, com despesas de entrega incluídas.

**Editora JOLAN Ltda.**  
Av. Rio Branco, 156 - sala 1537 - Tel.: 262-4619 - CEP 20040  
Rio de Janeiro, RJ.





Edgar Athayde Meneghetti

Este jogo foi escrito para os micros da linha Sinclair que possuem 16 Kb de memória, pois ele ocupa cerca de 3 Kb e é totalmente elaborado em linguagem de máquina. Nele, você comanda o submarino Polaris em águas inimigas, onde o objetivo é fazer o maior número de pontos sem ser destruído.

Inicialmente, carregue o MICROBUG e crie uma linha REM com 2542 caracteres. Introduza os códigos hexadecimais a partir do endereço 16509. Após ter digitado tudo, basta dar RUN que o programa será gravado, aparecendo em seguida a abertura. Para iniciar o jogo, pressione NEWLINE ou o botão de tiro do joystick que a batalha irá começar!

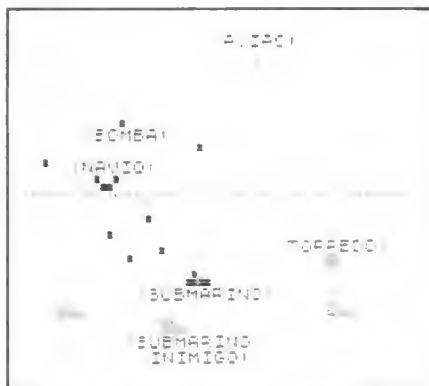
A tabela de pontos é mostrada na figura 1.

Avião = 10 pontos  
Barco = 50 pontos  
Submarino inimigo = 100 pontos  
Líder da tropa = 500 pontos  
Mina = não pode ser destruída  
Bomba = não vale pontos

Figura 1

16509	00 01 4E 01 EA 0E 4C 98	16757	36 00 18 F2 00 C9 C3 78	17005	11 08 00 19 36 12 19 36	17253	FE 02 38 01 C9 E5 11 DF
16517	80 CC 48 4A 4D 29 4D 08	16765	47 21 84 40 C8 7E 20 08	17013	12 19 36 12 C9 E5 05 2A	17261	FF 19 7E FE 00 20 02 36
16525	4D 7F 4A 8C 4A C1 FE 00	16773	23 34 C8 7E 20 01 C9 28	17021	32 40 54 5D 29 29 19 29	17269	88 E1 C9 2A 0C 40 01 F7
16533	C8 FE 08 C8 FE 89 C8 FE	16781	C8 FE C9 2A 86 40 28 7E	17029	29 29 19 22 32 40 7C D1	17277	02 07 06 04 28 7E FE 7A
16541	00 C2 84 44 C5 C9 2A 82	16789	FE 08 28 0D 22 86 40 36	17037	E1 C9 36 00 E5 11 21 00	17285	28 07 FE 09 CC 52 43 18
16549	40 36 B1 23 36 B3 CD 88	16797	86 23 36 06 23 36 00 18	17045	19 7E FE 09 28 F9 FE 89	17293	F3 10 FC C9 36 00 CD 7A
16557	02 23 C8 28 C8 4C C3 D8	16805	16 23 36 00 23 36 00 11	17053	28 18 FE 12 28 F2 FE 81	17301	42 FE 55 38 1C FE 8A 08
16565	49 C8 5C 28 11 C8 64 28	16813	18 00 19 22 86 40 21 84	17061	00 00 00 FE 83 00 00 00	17309	10 11 DF FF 19 7E FE 08
16573	24 C8 6C C0 C8 65 28 43	16821	40 C8 8E 23 36 00 C9 21	17069	FE 08 28 E4 FE 88 28 E0	17317	C8 FE 01 00 00 00 FE 83
16581	C8 5D 28 65 C9 00 2A 82	16829	86 40 C8 56 20 01 C9 2A	17077	36 02 E1 C9 2A 0C 40 01	17325	00 00 00 FE 80 C8 36 88
16589	40 23 23 7E CD 92 40 28	16837	86 40 11 42 00 19 36 02	17085	F7 02 09 06 15 28 7E FE	17333	C9 11 DE FF 18 E6 11 E8
16597	28 36 00 23 22 82 40 36	16845	C9 76 00 02 06 01 EA 2A	17093	76 28 07 FE 02 CC 0F 42	17341	FF 18 FD ED 78 78 40 CD
16605	81 23 36 B3 C9 2A 82 40	16853	0C 40 23 11 21 00 19 06	17101	F3 10 FC C9 00 00 00 00	17349	25 4A C3 34 48 00 00 00
16613	11 DF FF 19 7E CD 92 40	16861	20 36 80 23 10 FB 23 03	17109	00 00 00 76 00 03 FA 00	17357	00 00 00 00 00 00 00 00
16621	23 7E CD 92 40 28 22 82	16869	15 11 21 00 36 80 19 10	17117	EA 36 00 23 7E FE 80 28	17365	00 76 00 04 C4 00 EA 2A
16629	40 11 21 00 19 36 00 23	16877	F8 06 20 36 89 23 10 FB	17125	04 36 12 18 08 E5 11 E2	17373	0C 40 01 4A 01 09 06 00
16637	36 00 2A 82 40 36 81 23	16885	28 06 16 11 DF FF 19 36	17133	FF 19 36 12 E1 CD 7A 42	17381	23 7E FE 76 28 07 FE 88
16645	36 83 C9 2A 82 40 11 21	16893	00 10 FB 2A 0C 40 11 6D	17141	FE 02 38 01 C9 E5 11 21	17389	CC 91 43 18 F3 10 FC C9
16653	00 19 7E CD 92 40 23 7E	16901	01 19 06 1E 36 80 23 10	17149	00 19 7E FE 00 20 02 36	17397	FE 88 20 04 CD 5F 41 C9
16661	CD 92 40 28 22 82 40 11	16909	F8 2A 0C 40 11 9E 02 19	17157	02 E1 C9 2A 0C 40 01 8C	17405	36 00 F5 E5 CD 5F 41 E1
16669	DF FF 19 36 00 23 36 00	16917	22 82 40 2A 0C 40 11 96	17165	01 09 06 08 28 7E FE 76	17413	F1 FE 12 28 16 FE 09 28
16677	2A 82 40 36 81 23 36 83	16925	02 19 22 8C 40 11 21 00	17173	28 07 FE 12 CC DE 42 18	17421	18 FE 08 28 1D FE 06 28
16685	C9 2A 82 40 28 7E CD 92	16933	19 22 8A 40 19 22 88 40	17181	F3 10 FC C9 CD 7A 42 FE	17429	1E FE 06 28 1F 00 00 00
16693	40 22 82 40 23 36 00 23	16941	2A 0C 40 11 68 01 19 22	17189	F5 30 01 C9 CD 7A 42 CD	17437	36 00 C9 06 0A CD 5A 44
16701	36 00 28 36 B3 28 36 81	16949	86 40 2A 0C 40 11 0D 00	17197	7A 42 CD 7A 42 FE 55 38	17445	C9 28 36 00 06 4A CD 5A
16709	C9 00 2A 82 40 11 DF FF	16957	19 22 8E 40 2A 0C 40 11	17205	12 FE AA 38 13 2A 88 40	17453	44 C9 23 36 00 18 F5 23
16717	19 7E FE 00 28 0C FE 08	16965	66 00 19 36 12 11 08 00	17213	23 7E FE 00 C0 36 09 28	17461	36 00 18 03 28 36 00 C8
16725	28 F6 FE 00 20 36 18	16973	19 36 12 19 36 12 19 36	17221	36 08 C9 2A 8A 40 18 F0	17469	C8 CD 5A 44 21 84 40 C8
16733	18 EE 0E 04 06 FF 10 FE	16981	12 11 2C 00 19 36 12 11	17229	2A 8C 40 18 F9 28 36 00	17477	8E 23 36 00 2A 84 40 23
16741	0D 20 F9 11 21 00 19 7E	16989	04 00 19 06 06 36 12 19	17237	23 36 00 23 7E FE 80 C8	17485	7E FE 00 28 02 18 FB 28
16749	FE 08 28 FA FE B1 28 04	16997	10 FB 11 28 00 19 36 12	17245	36 09 28 36 08 CD 7A 42	17493	28 22 86 40 C9 E5 2A 8E

## Polaris



Pode-se jogar pelo teclado ou pelo joystick. E quando aparecer a abertura, o programa poderá ser brechado pressionando SHIFT e BREAK simultaneamente.

Na figura 2 encontramos um resumo dos principais endereços para eventuais modificações ou até mesmo uma personificação do jogo. Nela, observamos ainda que o número de bombas é a proporção de 0 a 254.

Se você quiser modificar o recorde, altere os códigos de 18021 a 18025;

- 18985 — Número de bombas lançadas pelos aviões (quanto maior o número mais bombas).
- 18990 — Número de submarinos (quanto maior o número mais submarinos).
- 18995 — Número de minas soltas pelos submarinos (quanto menor o número mais minas).
- 19000 — Número de bombas soltas pelo líder (quanto maior o número mais bombas).
- 17441 — Valor atribuído aos aviões.
- 17450 — Valor atribuído aos submarinos.
- 17469 — Valor atribuído aos barcos.
- 16736 — Valor da pausa entre a subida do tiro e o seu apagamento (caso o tiro esteja pouco visível, aumente o valor para 6 ou mais).

Figura 2

e para mudar o número de vidas, modifique o valor do endereço 17898 da seguinte forma: ao número de vidas desejadas, some 28.

Edgar Athayde Meneghetti cursa a segunda série do Colégio de Aplicação, de Porto Alegre. É um autodidata que aprendeu a programar em BASIC e Assembler através da leitura de livros e revistas.

Polaris

17501	40 7E FE 25 28 05 34 10	17909	11 12 00 19 44 4D 21 64	18317	FD 10 F9 2A 0C 40 11 21	18717	80 80 80 80 80 80 80 80
17509	F5 E1 C9 36 1C 28 7E FE	17917	46 23 03 7E FE 00 28 03	18325	00 19 06 16 23 7E FE 76	18725	80 80 80 80 80 80 80 80
17517	25 28 F8 34 10 E8 E1 C9	17925	02 18 F6 2A 0C 40 11 1A	18333	28 04 36 00 18 F6 10 FC	18733	80 80 80 80 80 80 80 80
17525	2A 82 40 7E FE 81 20 07	17933	00 19 22 90 40 CD 04 41	18341	06 FF CD 5A 44 06 F5 CD	18741	80 80 80 80 80 80 80 80
17533	23 7E FE 83 20 01 C9 2A	17941	CD A3 40 CD C2 44 18 FB	18349	5A 44 C3 03 47 00 98 98	18749	80 A6 80 80 80 80 80 80 80
17541	90 40 7E FE 1C 28 04 35	17949	2A 0C 40 11 07 00 19 E5	18357	98 AA 88 89 AA 80 85 87	18757	85 80 80 80 80 80 80 80
17549	C3 FF 44 C3 B1 45 00 C9	17957	11 08 00 19 44 4D E1 23	18365	84 A8 87 A6 82 A6 80 85	18765	80 80 80 80 80 80 80 80
17557	00 00 00 00 00 00 00 00	17965	03 7E 00 00 00 00 FE 00	18373	AA 87 89 AA 83 A8 AA 80	18773	80 80 80 80 80 80 80 80
17565	00 76 00 05 18 01 EA 2A	17973	28 12 0A 5E 88 28 F0 30	18381	AC 98 98 98 80 00 AA 89	18781	80 80 80 80 80 80 80 80
17573	0C 40 06 09 23 7E FE 76	17981	08 78 02 23 03 7E FE 00	18389	AC A6 87 80 82 AA 83 AA	18789	80 80 80 80 80 80 80 80
17581	28 05 FE 12 C8 18 F5 10	17989	28 02 18 F6 2A 0C 40 11	18397	AC AD AA 89 89 AE 80 90	18797	80 80 80 80 80 80 80 80
17589	FC C3 00 4A 36 00 23 36	17997	12 00 19 01 64 46 23 03	18405	A8 91 AA 82 80 9F 9D 98	18805	AC 80 80 80 80 80 80 80 80
17597	00 CD 04 41 C9 CD 75 44	18005	7E FE 00 28 03 02 18 F6	18413	9C A1 98 A4 A0 76 00 08	18813	80 AA 80 83 80 AA 80 AC
17605	CD A3 40 CD 7E 41 CD 08	18013	C3 27 48 AA A9 AC A6 87	18421	13 00 F8 08 85 84 B1 A6	18821	80 AD 80 AA 80 89 80 89
17613	43 CD DC 43 CD 89 42 CD	18021	1C 1C 21 1C 1C 1C 00 00	18429	87 AE 88 88 AA A9 AC A6	18829	80 AE 80 80 80 80 80 80
17621	75 44 CD A3 40 CD 7E 41	18029	00 00 2A 0C 40 11 44 00	18437	87 80 A4 A1 76 00 09 10	18837	80 80 80 80 80 80 80 80
17629	CD 08 43 CD 78 43 CD 21	18037	19 22 C4 4A 2A 0C 40 11	18445	00 F1 31 14 D4 1C 76 76	18845	80 80 80 80 80 80 80 80
17637	43 CD 75 44 CD 43 40 CD	18045	44 00 19 22 77 46 2A 77	18453	76 76 7E BF 10 46 00 00	18853	80 80 80 80 80 80 80 80
17645	7E 41 CD 08 43 CD 89 42	18053	46 36 00 23 36 07 00 00	18461	76 00 0A D9 01 EA C3 47	18861	80 80 80 80 80 80 80 80
17653	CD AA 44 CD 75 44 CD A3	18061	00 00 00 00 23 36 84 23	18469	4A 00 06 FF 0E FF 0D 20	18869	80 80 80 80 80 85 87 AA
17661	40 C9 06 20 2A 82 40 36	18069	7E FE 80 20 15 06 92 28	18477	FD 10 F9 C3 C0 43 00 2A	18877	80 80 80 80 80 80 80 80
17669	81 23 36 83 C5 2A 82 40	18077	36 00 10 F8 06 05 23 10	18485	0C 40 11 6F 48 13 23 7E	18885	AA BC 98 B1 AE 83 AA 80
17677	01 DF FF 11 21 00 23 7E	18085	FD 22 77 46 7E FE 08 28	18493	FE 76 28 FA 1A FE FF 28	18893	80 80 80 FF ED 73 78 40
17685	C6 80 77 28 7E C6 80 77	18093	18 C9 28 28 00 00 22 77	18501	03 77 18 F1 CD 88 02 C8	18901	C3 34 48 06 07 20 0F
17693	28 7E C6 80 77 19 7E C6	18101	46 CD 7A 42 FE 10 38 01	18509	4C 20 F9 C8 44 C8 2A 0C	18909	21 DA 49 34 7E FE 08 28
17701	80 77 23 7E C6 80 77 23	18109	C9 11 21 00 19 7E FE 08	18517	40 23 22 0E 40 2A 0C 40	18917	01 C9 36 00 C3 46 41 E5
17709	7E C6 80 77 23 7E C6 80	18117	C8 36 02 C9 E1 E1 21 03	18525	06 17 23 7E FE 76 28 04	18925	21 DA 49 36 07 E1 C3 86
17717	77 09 7E C6 80 77 09 7E	18125	47 E5 C3 84 44 44 00 00	18533	36 00 18 F6 10 FC C3 C0	18933	40 76 76 76 76 76 00 08
17725	C6 80 77 28 7E C6 80 77	18133	00 76 00 07 18 01 EA CD	18541	45 80 80 80 80 03 03 80	18941	73 00 EA CD 4D 4A 34 7E
17733	28 7E C6 80 77 28 7E C6	18141	79 46 CD 83 46 CD 89 42	18549	80 03 03 03 80 03 03 80	18949	FE 09 28 36 21 F6 42 34
17741	80 77 06 0A 0E FF 0D 20	18149	CD 75 44 CD A3 40 00 00	18557	80 03 03 03 80 03 03 80	18957	34 21 25 43 35 35 35 35
17749	FD 10 F9 C1 10 02 18 03	18157	00 CD 78 43 CD DC 43 CD	18565	80 07 03 03 80 03 03 80	18965	21 66 43 34 21 8A 46 34
17757	C3 09 45 2A 0C 40 01 94	18165	7E 41 CD 83 46 C9 CD DC	18573	80 80 80 80 80 85 80 86	18973	34 34 34 34 C3 FB 46
17765	02 09 06 03 23 7E FE 76	18173	46 CD DF 46 18 FB 2A 0C	18581	80 85 80 85 80 85 80 86	18981	21 F6 42 36 02 21 25 43
17773	28 0F FE 08 20 09 23 7E	18181	40 23 11 21 00 19 06 20	18589	80 85 80 85 80 85 80 86	18989	36 F5 21 66 43 36 02 21
17781	FE 09 28 03 28 36 00 10	18189	36 80 23 10 FB 23 06 15	18597	80 80 85 80 80 85 80 86	18997	8A 46 36 10 21 46 4A 36
17789	E8 10 FC C9 2A 0C 40 11	18197	11 21 00 36 80 19 10 FB	18605	80 80 80 80 80 85 80 86	19005	00 C9 CD 25 4A C3 FB 46
17797	76 01 19 36 A8 23 36 AE	18205	06 28 36 89 23 10 FB 28	18613	80 85 80 85 80 85 80 86	19013	00 00 CD 25 4A C3 D1 49
17805	23 36 82 23 36 80 23 36	18213	06 16 11 DF FF 19 36 80	18621	80 02 03 01 80 85 80 86	19021	3A 46 4A C6 9D 2A 0C 40
17813	A9 23 36 AA 23 36 80 23	18221	10 FB 2A 0C 40 11 6D 01	18629	80 80 85 80 80 02 03 03	19029	11 20 00 19 77 21 46 4A
17821	36 AF 23 36 A6 23 36 AC	18229	19 06 1E 36 80 23 10 FB	18637	80 80 80 80 80 87 83 80	19037	C9 18 18 18 18 18 18 18
17829	23 36 84 E1 E1 C3 1D 46	18237	2A 0C 40 11 66 00 19 36	18645	80 85 80 85 80 85 80 86		
17837	00 00 00 00 00 00 00 00	18245	12 11 08 00 19 36 12 19	18653	80 85 80 85 80 87 04 80		
17845	00 00 00 00 00 76 00 06	18253	36 12 19 36 12 11 2C 00	18661	80 80 85 80 80 80		
17853	18 01 EA 18 0C E1 7E 23	18261	19 36 12 11 04 00 19 06	18669	80 80 80 80 80 85 80 80		
17861	E5 FE 43 C8 CD 08 08 18	18269	06 36 12 19 10 FB 11 28	18677	80 02 03 01 80 85 03 03		
17869	F4 CD C2 45 35 34 33 39	18277	00 19 36 12 11 08 00 19	18685	80 85 80 80 80 85 80 86		
17877	34 38 00 1C 1C 1C 1C 1C	18285	36 12 19 36 12 19 36 12	18693	80 07 01 03 80 03 03 01		
17885	1C 00 07 2A 28 18 1C 1C	18293	C3 15 46 FE 07 28 08 FE	18701	80 80 80 80 80 80 80 80		
17893	1C 1C 1C 1C 00 1F 00 81	18301	03 28 07 FE 8A 28 0D C3	18709	80 80 80 80 80 80 80 80		
17901	83 00 00 9C 43 2A 0C 40	18309	43 03 46 AA 0E FF 0D 20				



Aldo Barduco Jr. e Pierluigi Piazzi

## O peixe guloso

No fim do dia, quando os pescadores se retiram, deixam latas com minhocas e outras iscas na beira do lago. Water, um peixe de nome científico *PISCES FAMINTUS*, está com muita fome e sai à procura de alimento. Tendo o hábito de ficar soltando bolhas na água, descobre que, ao soltá-las sob as latinhas, destas caem minhocas.

Os peixes desta espécie habitam os frios lagos do Tibet, onde existe o terrível *LUPUS AQUATICUS*, popularmente conhecido como Peixe-lobo. Os Peixes-lobos têm como dieta tudo que encontram pela frente, menos as terríveis Piranhas Azuis.

O único peixe capaz de se alimentar

destas Piranhas é Water, mas, se a Piranha estiver por trás, o devorará em poucos instantes (a "ecologia" da região é realmente confusa!).

O Tibet tem um relevo jovem (terciário), por isso são tão comuns desabamentos e terremotos. A partir de uma certa hora, os desabamentos dos barrancos do lago começam e não param mais.

Como os gatos têm sete vidas, Water possui três.

Boa sorte e muito cuidado. No Tibet, os mistérios são muitos. .

### O JOGO

Você deverá conduzir Water, desviando-o dos Peixes-lobos (e da Piranha, se

estiver atrás). As teclas para controle de Water são: 7 (para subir); 6 (para descer) e 0 (para soltar bolhas). No topo da tela, estão indicados os pontos e as vidas.

As bolhas só podem ser soltas uma de cada vez. Caso a bolha atinja a latinha, minhocas cairão na água, e Water deverá pegá-las. Já, se ela atingir o louco Monge Nadador, esta sumirá.

Isas amarelas de formas estranhas aparecerão na tela. Ao pegá-las, você adiciona 1000 pontos ao seu placar.

Os Peixes-lobos virão à sua frente em ziguezague. Desvie, pois, se não o fizer, a morte será certa. Já as Piranhas Azuis vêm por trás e tentam pegá-lo. Se você,

porém, pegá-la por trás, fará muitos pontos.

Ao passar dos 1000 pontos, uma pedra começará a cair da borda do lago e, se atingir Water, este perderá uma vida. Ao final, o programa mostrará os recordistas. Caso você tenha batido algum recorde, deverá introduzir seu nome, que será exposto à admiração pública e exaltação generalizada!

### PARTES IMPORTANTES DO PROGRAMA

Inicialmente, explicaremos as rotinas mais interessantes e importantes do programa Water e, a seguir, daremos algumas dicas de alterações.

A linha 10 faz a leitura dos DATA, que estão a partir da linha 9100 e coloca seus valores na memória do UDG redefinível pelo usuário (dizemos "redefinível pelo usuário", pois existem o UDG 1 e o UDG 0 que o próprio TK define).

A linha 11 dimensiona R\$ para que guarde o recorde e os recordistas.

Na linha 30, o primeiro PRINT executa uma impressão na parte inferior da tela, nas linhas de edição.

Na linha 50, iniciamos o movimento do Water, pois fazemos um LOOP de 2 até 31 e armazenamos o valor em x (que corresponde à coordenada x da posição do peixe). Nesta mesma linha, colocamos em O o valor 1, se O for 0, e 0, se O for 1. Daqui a pouco você verá o porquê.

A linha 70 faz a leitura do teclado.

Na linha 71, verificamos se existe algum caráter de cor diferente do fundo, à frente de Water. Este recurso nos permite saber qual caráter está na dada po-

sição (ver da linha 1000 até a 1025).

Na linha 75, verificamos se Water mudou de linha (subiu ou desceu) e, em caso afirmativo, apagamos a velha posição que ele ocupava.

Na linha 80, imprimimos Water e variamos sua cauda de acordo com o valor de O (veja linha 50).

A linha 85 verifica se estamos quatro caracteres atrás da piranha e, se estivermos, faz com que Water abra e feche a boca. Já a linha 101 verifica se a pedra atingiu Water.

Da linha 105 até a linha 210 há o movimento de todos os outros elementos na tela e a verificação de se algum destes atingiu Water. Não entramos em detalhes, pois os processos já foram mencionados (verificação de uma posição no vídeo, verificação nas próprias variáveis que contêm a posição do Water).

Da linha 2000 até a 2040 há a exposição dos nomes recordistas, verificando se houve recorde batido. Em caso positivo, é feita a inserção do nome no novo recordista. As linhas 9000, 9010 e 9020 são continuação desta rotina (2000-2040), sendo apenas de recursos sonoros e visuais do final do programa.

Na linha 3000, usamos recursos de desenho (PLOT, DRAW), para escrever "W" e desenhar a lua.

Na linha 3060, guardamos, na variável O\$, a melodia que toca durante a apresentação.

Na linha 3070, é impressa a assinatura do autor do jogo e o ano em que foi feito (Aldo 85).

As possíveis mudanças que não alterariam a estrutura do programa são:

Modificar as teclas para 8 e 9, ao in-

vés de 6 e 7, seria muito interessante para os que possuem joystick.

Para modificar o som do programa, altere a linha 95.

Se achar que o número inicial de vidas é muito pouco, altere o FOR V=3 ... para FOR V=Número de vidas iniciais desejado...

Não mude a cor do jogo, pois isto requer um conhecimento maior sobre o funcionamento das funções ATTR que, no caso, serão totalmente alteradas (em termos de valores referidos aos caracteres do jogo).

Para se gravar o programa, basta dar o comando direto SAVE "WATER" LINE 10 e o mesmo irá entrar em auto-start. Se quiser gravá-lo de uma forma mais compacta, execute a linha 10 e depois elimine as linhas de 9100 até o final e substitua a linha 10 por LOAD "" CODE. Em seguida, grave o programa em duas etapas:

- 1) SAVE "WATER" LINE 10
- 2) SAVE "WATER" CODE USR "a", 168

Quanto ao resto, deixamos por conta de sua habilidade e imaginação. Bom divertimento!

Aldo Barduco Jr. está cursando a 8ª série do primeiro grau e é autodidata em BASIC e Assembler. Possui um TK 85 e um TK90X.

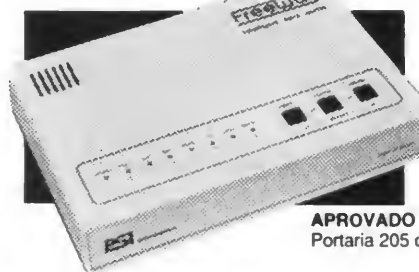
Pierluigi Piazzi é formado em Química e Física pela USP. Foi professor do curso Anglo Vestibulares e já editou a revista "Microhobby". É autor de vários livros e atualmente dirige a Aleph Publicações, responsável por muitos sucessos na área editorial, dada a sua orientação didática.

```

51REM Aldo Barduco Jr
10 FOR F=USR "a" TO USR "u"+7:
READ M: POKE F, M: NEXT F
11 LET A$="": LET B$="": FOR F
=1 TO 32 STEP 2: LET A$=A$+CHR$
140: LET B$=B$+CHR$ 149:CHR$ 148
: NEXT F: LET A$=A$+A$: DIM R$(1
0,20): DIM R(10)
12 BORDER 1: PAPER 1: INK 7: G
OSUB 3000
20 BORDER 1: CLS: LET Y=19: L
ET BX=0: LET BY=0: LET BO=0: LET
O=0: LET NX=0: LET NY=3: DIM M(
3): LET P=0: LET T=0: LET TC=0:
LET V=0: LET TY=0:
30 PRINT #1: " BRIGHT 1:CH
R$ 144:CHR$ 145: BRIGHT 0: Wate
r, o peixe guloso": LET MY=4: LE
T MX=0: PRINT AT 21,0:AT 21,0
: OVER 1:85
40 LET K=0: LET PX=0: LET PY=1
0: LET DX=0: LET DY=10: FOR F=0
TO 2: PRINT AT F,0: PAPER 5:
NEXT F
45 LET TY=Y: PAPER 5: INK 0: P
RINT AT 1,5:CHR$ 146:AT 1,15:CHR
$ 146:AT 1,25:CHR$ 146:AT 2,5:CH
R$ 147:AT 2,15:CHR$ 147:AT 2,25:
CHR$ 147: PAPER 1: INK 7
46 PRINT AT 0,3: PAPER 5: INK
0: "Pontos": PAPER 5: INK 0:AT 0
,10:P
49 FOR V=3 TO 1 STEP -1: PRINT
AT 0,14: PAPER 5: INK 0: "Vidas
1": PAPER 5: INK 0:V: SOUND 10,2
0: PRINT AT 3,0:
50 FOR X=2 TO 31: LET O=0+(O=0
)-(O=1)
70 LET Y=Y+(INKEY$="6") AND Y(1
9)-(INKEY$="7") AND Y(5): IF NOT
BO THEN LET BO=(INKEY$="0"): LET
BY=(Y-1 AND BO): LET BX=(X-2 AND
BO): IF BO THEN PRINT AT BY,BX
:CHR$ 151
71 LET K=K+(ATTR (Y,X+1):(15)
75 IF TY(Y) THEN PRINT AT TY,X
-3: " " : LET TY=Y
80 PRINT AT Y,X-2: " +(CHR$ 14
4+CHR$ 145 AND O=1)+(CHR$ 150+CH
R$ 145 AND O=0)
85 IF PX=4X AND X(PX THEN PRI
NT AT Y,X-2: " :CHR$ 144:CHR$ 15
90 IF K THEN LET K=0: GOTO 100
95 SOUND 100,12
100 IF P(1000 THEN PRINT AT MY,
MX: " : LET MY=MY+(MY(20)-(MY-4
AND MY(19)): PRINT AT MY,MX:CHR$
150: SOUND 100,15
101 IF (MX=X AND MY=Y) OR (MX=X
-1 AND MY=Y) THEN PRINT AT Y,X-1
: " : NEXT V: GOTO 2000
105 IF MY=4 THEN PRINT AT MY,MX
: " : LET MX=INT (RND*31)
110 IF M(1) AND (PX=X-1 AND PY=
Y) THEN PRINT AT Y,X-2: " : NE
XT V: GOTO 2000
120 IF T THEN LET T=T-1
130 IF BY=NY AND BX=NX THEN LET
BO=0
140 LET NX=NX-(NX)0+(31 AND NX
=0): PRINT AT NY,NX:CHR$ 157 AND
DX/2=INT (NX/2)+(CHR$ 156 AND
NX/2=INT (NX/2): "
150 IF BO THEN PRINT AT BY,BX: "
: LET BY=BY-(BY)3: PRINT AT B
Y,BX:CHR$ 150
160 IF M(1) THEN PRINT AT PY,PX
: " : LET PX=PX+(INT (RND*2) AND
PX(31)-(30 AND PX)30): LET PY=P
Y+SGN (Y-PY)+INT (RND*2): PRINT
AT PY,PX: INK 5:CHR$ 153
170 IF M(2) THEN PRINT AT DY,DX
: " : LET DX=DX-1: LET DY=DY+SGN
(Y-DY-RND*1.5): PRINT AT DY,DX:
INK 4:CHR$ 155: IF NOT DX THEN
PRINT AT DY,DX: " : LET M(2)=0
180 IF NOT M(1) AND RND(.1) THEN
LET PX=0: LET M(1)=PI/PI
195 IF NOT M(3) AND RND(.02) THE
N LET M(3)=1: PRINT AT RND*15+5,
RND*29+1: INK 6:CHR$ 154
200 IF NOT M(2) AND RND(.2) THEN
LET DX=31: LET M(2)=PI/PI
205 IF (DX=X-1 AND DY=Y) OR (DX
=X AND DY=Y) THEN PRINT AT Y,X-2
: " : LET DY=(21-4): NEXT V: G
OTO 2000
206 IF BY=3 THEN LET BY=2: LET
BO=0: PRINT AT 3,BX: " : GOTO 0+
(900 AND ATTR (BY,BX)=40)+(207 AND
ATTR (BY,BX)=140)
207 IF BY=3 THEN PRINT AT BY,BX
: " : LET BO=0
208 IF NOT NX THEN PRINT AT NY,

```

## Freeway data buffer



APROVADO PELA SEI  
Portaria 205 de 23-08-85

ISI informática

Avenida Pavão, 346 - Indianópolis - CEP 04516 - São Paulo - SP  
Telefone: (011) 533-0120 / 533-0112 - Telex: 22966

**Aumente a produtividade do seu micro, acabando com a espera pela impressora.**

FREEWAY é um "data buffer" que controla sozinho a impressora durante a emissão dos seus relatórios, liberando o micro para outras tarefas.

- Memória de 64 Kb expansível a 576 Kb.
- Opera em modo comprimido, permitindo armazenar relatórios muito maiores que sua memória.
- 9 funções: COPIAS ADICIONAIS SEM USO DO MICRO, AVANÇO DE PÁGINA, PAUSA AUTOMÁTICA POR PÁGINA, ESTATÍSTICAS, MODO DIRETO, AUTOTESTE, etc.
- Comunicação serial (RS 232), paralela (Centronics) e em impressora paralela e vice-versa.
- Pode ser ligado em terminais, plotters, RJE, etc.



# Rally

Luís Maurício Chopard Bonilauri

Rally/Bas é um jogo rápido e emocionante, que poderá diverti-lo por longas horas. Ele foi programado em BASIC, mas possui duas rotinas em linguagem de máquina, para agilizá-lo e produzir som.

Originalmente, o programa foi feito em um CP 500, com drive, e para rodá-lo em equipamentos com cassete é necessário uma modificação, já que ele possui duas rotinas em Assembler, e o monitor BASIC residente só suporta uma. A modificação consiste em suprimir a rotina de som, uma vez que a outra rotina serve para deslocar a tela, sendo por isso mais importante. Deve-se proceder da seguinte maneira:

1º) Alterar a linha 80 para: **80 POKE 16526,0:POKE 16527,125.**

2º) Retirar as linhas 120, 130 e 180.

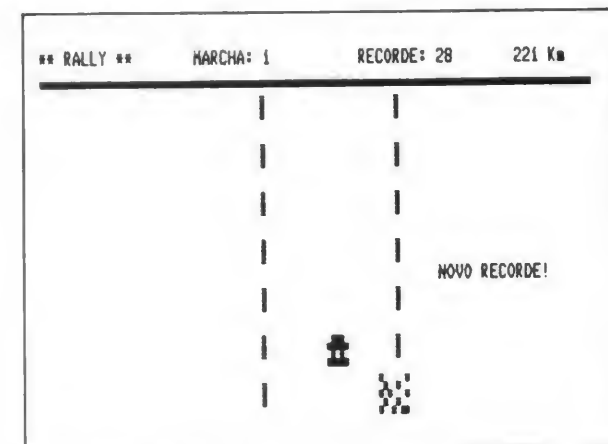
3º) Retirar da linha 340 o trecho: **FOR Y=1 TO 5:X=USR1(50):NEXT.**

4º) Mudar a linha 470 para 470 REM.

```

10 *****
20 * LUIS MAURICIO CHOPARD BONILAURO, RIO DE JANEIRO - R.J. *
30 *****
40 CLEAR 300
50 RE=0
60 RANDOM
70 A%=CHR$(174)+CHR$(175)+CHR$(132)+CHR$(26)+STRING$(3,B)+CHR$(
71)+CHR$(131)+CHR$(129)
80 DEFUSR=32000:DEFUSR1=-8192
90 B%=CHR$(170):C%=CHR$(149)
100 F=32000
110 READ A:POKE F,A:F=F+1:IF A()201 THEN GOTO 110
120 F=-8192
130 READ A:POKE F,A:F=F+1:IF A()201 THEN GOTO 130
140 GOSUB 400
150 GOSUB 500:CLS:PRINT@0,"** RALLY **"      MARCHE: 1
RECORDE: 28
160 P=925:M=1:I=0
170 DATA32,127,63,126,17,255,63,18,1,0,3,237,184,33,154,60,54,
,17,155,60,1,14,0,237,176,33,217,60,54,32,17,218,60,1,14,0,237,
76,201
180 DATA205,127,10,77,68,62,1,105,211,255,45,32,253,60,105,11,
55,45,32,253,13,16,238,175,211,255,201
190 GOSUB 370
200 PRINT@25+128,B%:@41+128,C%:
210 <=USR(0)
220 IF PEEK(15350)=A4P=P+2 ELSE IF PEEK(15350)=16P=P-2
230 IF PEEK(15360+P)128 OR PEEK(15362+P)128 GOTO 310
240 PRINT@P,A%:
250 IF PEEK(14352)=2 THEN NM=1:PRINT@26,"1":ELSE IF PEEK(14352)=
4 THEN NM=0:PRINT@26,"2":
260 IF RND(V)=1 PRINT@P-758,A%:
270 I=I+1
280 IF M=0 THEN GOTO 200
290 FOR F=1 TO 25:NEXT
300 GOTO 200
310 PRINT@54,I:"Km":
320 PRINT@92,STRING$(15,128):@986,STRING$(15,128):
330 PRINT@P-64,CHR$(144):" "CHR$(144):@P,CHR$(166):I HP$(132):
HR$(145):CHR$(132):@P+64,CHR$(129):CHR$(130):CHR$(130):CHR$(130):
:
340 IF I)RE THEN RE=I:PRINT@760-128-10,"NOVO RECORDE: "RE:
TO 5:X=USR1(50):NEXT ELSE FOR C=1 TO 300:NEXT
350 PRINT@898,"TECLE (ENTER) P/":@965,"RECORDEAR":
360 IF PEEK(15350)=0 THEN 360 ELSE GOTO 150
370 FOR F=2 TO 15 STEP 2:PRINT@F+64+25,CHR$(170):@F+64+41,CHR$(1
49):NEXT:PRINT@P,A%:
380 FOR F=0 TO 20:PRINT@54,"ATENCAO "FOR I=1 TO 35:PRINT@54,
4,STRING$(7,128):FOR I=1 TO 34:NEXT:NEXT
390 RETURN
400 CLS:PRINT@23,"* * R A L L Y * *":@87,STRING$(19," ")
410 PRINT:PRINT:PRINT" Dirija seu carro em uma pista de Rally,
mas cuidado com os outros concorrentes!"PRINT
420 PRINT" Use ( e ) para movimentar seu carro, e 1 e 2 para en
gatar a primeira ou segunda marcha, respectivamente. PRINT
430 PRINT" BOA SORTE!!"
440 PRINT:PRINT:PRINT" Por: Luis Mauricio Chopard Bonilauri
450 PRINT@977,"TECLE (ENTER) PARA COMEÇAR A CORRIDA":
460 F=30
470 X=USR1(30030+F):F=F+10:IF F=150 THEN I=0
480 IF PEEK(15350)=0 THEN 470
490 RETURN
500 CLS:PRINT@519,"Qual o nivel (dificil: 1,2,3,4,5,6,7,8,9,10):
510 IF V)21 OR V<5 THEN 500
520 RETURN

```



Exemplo de tela do jogo.

Luís M. C. Bonilauri tem 13 anos e, desde 1984, faz programas em BASIC e Assembler. Aprendeu as duas linguagens através de livros e manuais.

Rally/Bas

# O UNITRON REVELA O SEU SEGREDO:



## MODULARIDADE

Quem compra um micro está investindo no presente e no futuro. Por isso o equipamento deve acompanhar a evolução das necessidades e aplicações do usuário. Através dos seus oito slots (parte superior da foto), o UNITRON APII permite a expansão do equipamento na medida em que crescem as necessidades e de acordo com as economias de cada um.

Você pode começar comprando um UNITRON APII e ligá-lo na TV (P&B ou cores). Depois pode acrescentar disk-drives, impressora, monitor profissional, placa Z-80 para CP/M, placa 80 colunas, interface de co-

municações, etc., etc. Você pode também aumentar a memória do seu micro com placas de 16, 32, 64 ou 128K. Tudo isto de modo bastante simples, com manuais em português e fácil instalação. Quanto ao software, você dispõe de programas como o VISICALC, DBASE, WORD STAR e milhares de outros best-sellers, desenvolvidos para o "APPLE II+" norte-americano. Essa modularidade e riqueza de software é que fazem do UNITRON APII o micro mais vendido para empresas, profissionais liberais e famílias inteligentes. Fique com o melhor. Fique com o UNITRON APII.

### O MELHOR CONJUNTO

UNITRON APII, MONITOR COMPO, DRIVE ELEBRA, IMPRESSORA MONICA E INTERFACES.

**2690RTN's**

válido até 30/01/86

assessoria

O MAIOR REVENDEDOR UNITRON DE TODO O PAÍS

RUA SETE DE SETEMBRO, 99 - 11º ANDAR  
TEL.: PABX (021) 224-7007 - CEP 20050 - RIO DE JANEIRO - RJ  
AV. PAULISTA, 726 - 5º ANDAR - CONJ. 506  
TEL.: (011) 288-4492 - CEP 01362 - SÃO PAULO - SP

**compu**micro  
INFORMÁTICA EMPRESARIAL LTDA





Crie seus próprios jogos e movimente figuras em alta resolução, nos micros da linha Apple, utilizando as rotinas da ROM.

## Alta resolução em Assembler (I)

Evandro Mascarenhas de Oliveira

**D**e maneira análoga ao publicado em MS nº 27, serão discutidos os aspectos ligados ao uso das sub-rotinas da ROM e endereços da página zero para a geração e movimentação de figuras em alta resolução gráfica, utilizando a linguagem de máquina do microprocessador 6502.

A alta resolução gráfica ocupa duas páginas de 8 Kb da memória de vídeo, indo a página 1 do endereço \$2000 ao endereço \$3FFF e a página 2 do endereço \$4000 ao endereço \$5FFF.

A página do modo texto é constituída por 24 linhas e 40 colunas, tendo como unidade gráfica a matriz de 7 colunas por 8 linhas. Em modo gráfico de alta resolução, esta matriz é distribuída nas 24 linhas e 40 colunas resultando em  $24 \times 8 = 192$  linhas em modo gráfico completo, ou  $20 \times 8 = 160$  linhas em modo gráfico/texto e  $40 \times 7 = 280$  colunas, num total de  $280 \times 192 = 53760$  pontos.

Para diminuir a quantidade da memória de vídeo, relativa aos 53760 pontos, adotou-se um byte de 7 bits (o oitavo bit serve para controle da cor) para unidade gráfica por linha, ficando os 280 pontos em cada linha referenciados por  $280/7 = 40$  bytes ou endereços, havendo no total das linhas  $192 \times 40 = 7680$  endereços (os 512 endereços restantes não utilizados na memória de vídeo dão o total de 8192 bytes ou 8 Kb por página).

Quando um bit tem o valor 1, acende um ponto na tela; e quando tem o valor 0, mantém-se apagado. Se um determinado byte na memória de vídeo, por exemplo, tem o valor \$73 (binário 01110011), ter-se-á cinco pontos com nível alto — portanto, acesos — e dois com nível baixo — apagados — que se apresentam de maneira reversa ao binário (figura 1).

A determinação da posição de um ponto luminoso e a página da memória de vídeo onde está sendo exibido, estão relacionadas às seguintes sub-rotinas e endereços da página zero:

1 — HPOSN: Sub-rotina com início em \$F411, dá a posição inicial da linha para o ponto de coordenadas C (coluna) e L (linha), sendo que:

C — Varia de 0 a 279 (\$00 a \$117), cujos valores necessitam ser referidos por dois endereços, LSB e MSB, respectivamente,

byte menos significativo e byte mais significativo.

L — Varia de 0 a 191 (\$00 a \$BF), bastando um só endereço para ser referenciada.

Carregando o acumulador com o valor da linha, o registrador X com o LSB e o registrador Y com o MSB da coluna, ter-se-á o endereço inicial da linha nos endereços \$27(MSB) e \$26(LSB), designados, respectivamente, por HBASH e HBASL.

2 — HNDX: Representado pelo endereço \$E5, é o número de bytes a ser adicionado ao endereço inicial da linha. Indica ainda a parte inteira da coordenada (coluna) dividida por sete.

3 — HPAG: Dado pelo endereço \$E6, mostra a página da memória de vídeo onde está sendo exibido o ponto. Tem o valor \$20, se for página 1; e \$40, caso seja página 2.

4 — HMASK: Colocado no endereço \$30, indica a posição do bit ligado (valor 1) dentro do byte das colunas de 0 a 279. O byte zero, representativo das colunas de 0 a 6 (7 colunas),

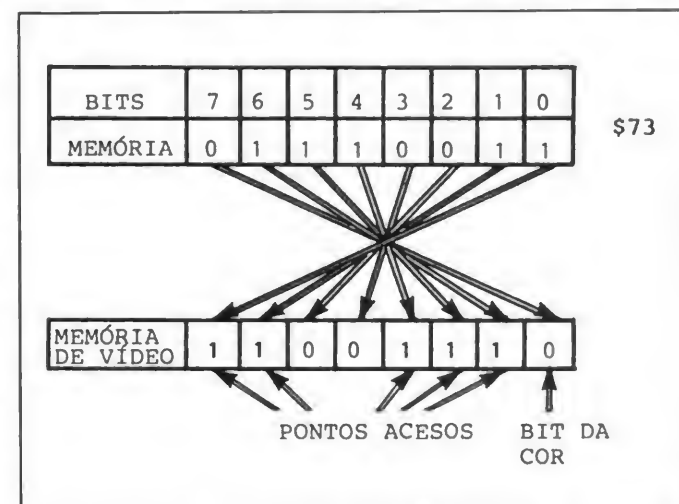


Figura 1 — Hexadecimal 73 e seu respectivo binário em algum endereço da memória, além de sua referência na memória de vídeo.

## NA CLAPPY, CRAFT II PLUS E CRAFT-XT COM 3 SOFTWARE EXCLUSIVOS.

### CRAFT II PLUS

CPU com teclado numérico reduzido. Monitor profissional Compo (24 MHz). Interface e drive 5' 1/4". 32 funções pré-pro-

gramadas e 10 funções programáveis. Sistema Ivanita (geração de caracteres em português). Conectores banhados a ouro.

Na Clappy, o Craft II Plus e o Craft-XT ainda têm 3 software exclusivos: o atendimento especializado, a assistência técnica perfeita e as melhores condições de preço e pagamento da cidade.

Passe na Clappy e faça um teste sem compromisso. Você vai ver como o Craft II Plus e o Craft-XT são altamente compatíveis com você.

**MICROCRAFT**  
MICROCOMPUTADORES LTDA.

**Clappy**

O lado gente da máquina

**Centro:** Av. Rio Branco, 12 - loja e sobreloja  
Tel.: (021) 253-3395

**Copacabana:** Rua Pompeu Loureiro, 99  
Tels.: (021) 236-7175/257-4398

Visite nossas lojas ou solicite a visita de um representante.



### CRAFT-XT

PC 16 bits IBM compatível. Memória RAM de 256 kbytes, expandível até 640 kbytes. Opera em tempo real. 2 drives de 5' 1/4" de 340 kbytes cada ou drive de 5' 1/4" de 340

kbytes e até 2 discos rígidos Winchester de 10 Mbytes cada. Teclado de 85 teclas com microprocessador. Monitor monocromático de 12", com tela verde anti-reflexiva.

tem os valores lidos do endereço \$30, conforme a figura 2. Estes valores se repetem para todos os 40 bytes de cada linha (byte \$00 ao byte \$27), na mesma sequência.

Linha	BITS EM POSIÇÃO REVERSA								C	L
	0	1	2	3	4	5	6	7		
0	1	0	0	0	0	0	0	1	8	1
1	0	1	0	0	0	0	0	1	8	2
2	0	0	1	0	0	0	0	1	8	4
3	0	0	0	1	0	0	0	1	8	6
4	0	0	0	0	1	0	0	1	9	0
5	0	0	0	0	0	1	0	1	A	0
6	0	0	0	0	0	0	1	1	C	0
	LSB				MSB				MSB LSB	

Figura 2 - Tabela de valores do endereço \$30 para o byte zero, relativo às colunas de 0 a 6. O bit 0 é lido da direita para a esquerda.

O programa a seguir exibe na tela de vídeo, página 1, o valor 2000 para o endereço do ponto com coordenadas C=3 e L=0 (MSB = 20, endereço \$27 e LBS = 00, endereço \$26) o valor 0 para o endereço \$E5 (parte inteira de 3/7 = 0, portanto, byte \$00) e o valor 88 no endereço \$30.

03001	1	DRG	\$0200	1	ENDEREÇO INICIAL DO PROGRAMA
0300120	2	JSR	\$FC5B	1	ILIMPA A TELA DE TEXTO
0300140	3	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300160	4	STA	\$E6	1	ILIMPA A TELA DE TEXTO
0300180	5	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03001A0	6	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03001C0	7	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03001E0	8	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300200	9	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300220	10	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300240	11	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300260	12	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300280	13	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03002A0	14	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03002C0	15	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03002E0	16	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300300	17	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300320	18	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300340	19	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300360	20	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300380	21	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03003A0	22	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03003C0	23	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03003E0	24	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300400	25	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300420	26	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300440	27	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300460	28	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300480	29	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03004A0	30	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03004C0	31	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03004E0	32	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300500	33	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300520	34	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300540	35	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300560	36	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300580	37	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03005A0	38	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03005C0	39	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03005E0	40	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300600	41	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300620	42	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300640	43	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300660	44	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300680	45	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03006A0	46	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03006C0	47	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03006E0	48	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300700	49	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300720	50	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300740	51	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300760	52	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300780	53	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03007A0	54	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03007C0	55	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03007E0	56	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300800	57	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300820	58	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300840	59	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300860	60	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300880	61	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03008A0	62	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03008C0	63	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03008E0	64	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300900	65	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300920	66	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300940	67	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300960	68	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300980	69	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03009A0	70	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03009C0	71	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
03009E0	72	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300A00	73	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300A20	74	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300A40	75	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300A60	76	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300A80	77	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300AA0	78	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300AC0	79	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300AE0	80	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300B00	81	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300B20	82	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300B40	83	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300B60	84	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300B80	85	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300BA0	86	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300BC0	87	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300BE0	88	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300C00	89	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300C20	90	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300C40	91	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300C60	92	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300C80	93	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300CA0	94	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300CC0	95	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300CE0	96	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300D00	97	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300D20	98	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300D40	99	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300D60	100	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300D80	101	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300DA0	102	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300DC0	103	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300DE0	104	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300E00	105	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300E20	106	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300E40	107	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300E60	108	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300E80	109	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300EA0	110	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300EC0	111	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300EE0	112	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300F00	113	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300F20	114	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300F40	115	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300F60	116	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300F80	117	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300FA0	118	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300FC0	119	LDA	\$E6	1	ILIMPA A TELA DE TEXTO
0300FE0	120	LDA	\$E6	1	ILIMPA A TELA DE TEXTO

- 5 - HCLR: Sub-rotina com início em \$F3F2, limpa a página de acordo com o valor em \$E6.
- 6 - HGR: Sub-rotina com início em \$F3E2, limpa a página 1 e coloca a tela em modo misto gráfico/texto.
- 7 - HGR2: Sub-rotina com início em \$F3D8, limpa a página 2 e coloca a tela em modo gráfico completo (full screen).
- 8 - BKGND: Sub-rotina com início em \$F3F4, gera o fundo com a cor nos endereços \$1C ou \$E4.
- O programa seguinte coloca a tela em modo misto, página 1, limpa a área de texto e coloca o fundo com a cor branca:

03001	1	DRG	\$0200	1	ENDEREÇO INICIAL DO PROGRAMA
0300120	2	JSR	\$FC5B	1	ILIMPA A TELA DE TEXTO
0300140	3	JSR	\$FCE2	1	ILIMPA A TELA DE TEXTO
0300160	4	LDA	\$E6F	1	ILIMPA A TELA DE TEXTO
0300180	5	STA	\$E4	1	ILIMPA A TELA DE TEXTO
03001A0	6	JSR	\$FCE4	1	ILIMPA A TELA DE TEXTO
03001C0	7	RTS		1	ILIMPA A TELA DE TEXTO





Envie suas dicas para a Redação de MICRO SISTEMAS na Av. Presidente Wilson, 165 - grupo 1210. Centro, Rio de Janeiro, RJ. CEP 20030

## Linha SHARP PC-1500

### Utilizando rótulos

Facilite a estruturação, a compreensibilidade e a manutenção de seus programas no PC-1500, fazendo uso de rótulos como no exemplo abaixo:

```
Ao invés de:
50 GOSUB 1000
1000 C = P * N: RETURN
```

```
Faça:
50 GOSUB "ROTINA DE
CUSTO"
1000 "ROTINA DE CUSTO"
C = P * N: RETURN
Também aceitam rótulos os
seguintes comandos:
GOTO, IF... THEN, LIST,
LLIST, RESTORE, RUN.
```

Aldo Rodrigues da Costa — MG

## Linha APPLE

### Mensagem em parábola

Esta dica serve para mostrar no vídeo a mensagem contida na variável A\$. Caso a mensagem exceda os limites da tela, basta modificar a fórmula da linha 30.

```
5 HOME: INPUT "HOME ?": AS: HOME
10 FOR A=0 TO 6.2 STEP .2
20 B=SIN(A)
30 PRINT TAB(B*10+15); A$
40 NEXT A
50 GOTO 10
```

Jorge Pablo Zapata — BA

## Linha TK 2000

### Amplie caracteres

Com esta dica, você poderá ampliar os caracteres do seu TK 2000. Digite as linhas abaixo e rode com RUN, depois apague-os:

```
1 DATA 165,9,162,7,173,0,64,133,
0,6,0,6,0,144,5,165,2,32,0,248,1
36,202,208,243,230,2,173,6,3,24,
105,4,141,6,3,201,64,200,217,96
2 FOR F=768 TO 807: READ P: POKE F
,P: NEXT F
```

Para usar o programa, digite:

```
10 HOME: INPUT "QUAL A COR ?
": AT: PRINT: INPUT "QUAL A LETRA ?
": AS
20 GET: POKE 48,A: HTAB 1: VTAB 1: PRI
NT AS
30 POKE 774,32: POKE 2,10
40 CALL 768
50 GET AS: TEXT: RUN
```

William Joseph Watterson — SP

## Linha ZX Spectrum

### Memória disponível

Para saber quanto resta de memória em seu micro, digite as linhas abaixo e rode o programa. Depois, sempre que quiser saber quanto sobra de memória, use PRINT USR 23300.

A rotina fica guardada na área do buffer da impressora, e se você quiser colocá-la em outro local da memória, basta mudar os endereços da linha 10 antes de rodar o programa.

```
10 FOR F=23300 TO 23310
20 READ A: POKE F,A: NEXT F
30 DATA 205,26,31,42,178,92,23,7
,66,68,77,201
```

Carlos Alberto Guerreiro — MG

## Linha ZX81

### Caleidoscópio

Aí vai uma Dica que explora a capacidade gráfica do seu compatível com ZX81. Após rodar o programa, mude o seu efeito alterando a linha 30, para:

```
30 LET AS=CHR$(INT (RND*10+128))
30 LET AS=CHR$(INT (RND*10+128)):
CHR$(INT (RND*10+1))
```

Podendo então alterar o comprimento da variável A\$.

```
1 REM -- CALEIDOSCOPIO --
10 CLS
20 LET A=1
30 LET AS=CHR$(INT (RND*10+1))
40 LET B=20
50 LET C=B*RND
60 LET D=B*RND
70 PRINT AT C,D: AS
80 PRINT AT C,D: AS
90 PRINT AT C,B-D: AS
100 PRINT AT B-D,C: AS
110 PRINT AT D,B-C: AS
120 PRINT AT B-C,D: AS
130 PRINT AT B-C,D: AS
140 PRINT AT B-D,B-C: AS
150 LET A=A+1
160 IF A=20 THEN RUN
170 GOTO 50
```

José Rogério Prates — SP

## Linha APPLE

### Impressão profissional

Esta dica serve para escrever em seus joguinhos ou aplicativos pequenas mensagens com mais elegância. A mensagem deve ser introduzida na variável A\$, que se encontra na linha 5. Para mudar a linha, onde vai ser impressa a mensagem, basta mudar o VTAB da linha 30.

```
5 HOME: INPUT "FRASE ?": AS: HOME
10 FOR A=1 TO LEN(AS)
20 VTAB 1: HTAB A
40 B$=MID$(AS,A,1)
45 SOUND ASC(B$)+20,4
50 PRINT B$: CHR$(242): CHR$(199)
60 NEXT A
```

Jorge Pablo Zapata Rivera — BA

## Linha ZX Spectrum

### UDG 2 do TK90X

A função UDG 2 do TK90X é uma *mão na roda* para redefinir os caracteres do micro, mas nem tudo foi dito no manual. Experimente apertar uma tecla de A a U, acompanhada de CAPS SHIFT e você verá que pode passar a redefinição da tecla pressionada para a letra que está em edição.

Ricardo Costa — RJ

## Linha ZX81

### Efeito na tela

Esta rotina provoca nos computadores da linha ZX81 um efeito muito interessante no vídeo, podendo ser usada também em qualquer programa:

```
10 HIM ..... (5 caracteres
quaisquer)
20 POKE 16514,62
30 POKE 16515,48
40 POKE 16516,237
50 POKE 16517,71
60 POKE 16518,201
70 RAND USR 16514
80 FOR A=0 TO 100
90 POKE 16515,A
100 RAND USR 16514
110 NEXT A
```

Você notará que após rodar a rotina, todos os caracteres estarão indefinidos; não se alarme, para voltar ao normal é só dar estes comandos diretos:

```
POKE 16515,30 (new line) e
RAND USR 16514
```

Quem Woo Moo — SP

## Linha APPLE

### Conversão DEC-HEX

Muitas vezes, se faz necessária uma rápida conversão Decimal-Hexadecimal. Normalmente, um programa para esse fim, além de ser lento, necessita de uma série de manipulações de "Strings" e é um pouco dispendioso.

Mas, neste programa, a conversão é feita usando uma rotina intrínseca do Monitor, o que nos garante uma alta velocidade e uma simplicidade maior.

Esta rotina "PRINTA" o conteúdo

dos registros XY em HEXA, e seu endereço é \$FD99.

Observação: O valor máximo decimal é 65535.

```
10 HOME
20 REM DEC-HEX
30 PRINT "ENTRE COM O NUMERO DEC
": INPUT Z: PRINT: PRINT
40 K=INT(Z/256): F=Z-K*256
50 X=F: Y=K
60 POKE 768,162: POKE 769,X: POKE
770,160: POKE 771,Y
70 POKE 772,32: POKE 773,153: POKE
774,253
75 POKE 775,96
80 CALL 768
90 PRINT: PRINT
100 GOTO 30
```

Alexandre Magno Rodrigues Alves — SP

## Linha ZX81

### Funções especiais

Use esta dica para descobrir o nome de um programa em seu TK85.

```
10 LET Z=0
20 LET Z$=""
30 DIM A$(100)
40 RAND USR 8305
50 FOR I=1 TO LEN A$
60 IF CODE A$(I)<128 THEN NEXT I
70 PRINT "PROGRAMA: "; A$(TO I)
```

Enrique Raul Broens — PR

## Linha TRS-80

### Destrave o LIST no TRS-80

Com o seguinte POKE, qualquer programa com LIST todo bagunçado poderá ser listado SOMENTE NA TELA. POKE 17129,239. Se não funcionar, desligue e ligue o computador, recarregue o programa e dê POKE 17130,66. Se ainda não funcionar, desligue e ligue o computador e vá direto para o BASIC residente, dê um PRINT PEEK (17129), PEEK (17130), anotando os valores impressos e, depois, entre no DOS. Carregue o mesmo programa e dê POKE 17129, n1: POKE 17130, n2, sendo n1 e n2 os valores que você tinha adotado.

Bruno Lopes Fernando Cabral — PB

## Linha APPLE e TK 2000

### Gravação de dados

Um dos grandes problemas do Applesoft/BASIC é a gravação de dados alfanuméricos em fita, o que dificulta muito a criação de programas do tipo Banco de Dados, por exemplo.

A solução é simular uma função que execute com precisão e segurança. É o que as rotinas abaixo propõem.

#### ROTINA 1:

```
10 HOME
20 CLEAR
30 DIM A(255), AS(255)
40 INPUT "TEXTO: ": B$
50 A(0)=LEN(B$)
60 FOR I=1 TO A(0)
70 AS(I)=MID$(B$,I,1)
80 NEXT I
90 FOR I=1 TO A(0)
100 A(I)=ASC(AS(I))
110 NEXT I
120 PRINT "PRONTU?": GET X$
130 STORE A
140 PRINT: PRINT "OK"
```

#### ROTINA 2:

```
10 HOME
20 CLEAR
30 DIM A(255), AS(255)
40 PRINT "PRONTU?": GET X$
50 RECALL A
60 FOR I=1 TO A(0)
70 AS(I)=CHR$(A(I))
80 NEXT I
90 PRINT "TEXTO: "
100 FOR I=1 TO A(0): PRINT AS(I):
NEXT I
```

Estes dois pequenos programas rodam sem problema no TK 2000. É ACONSELHÁVEL, porém, que estes sejam depurados para funcionarem perfeitamente, de acordo com as necessidades de cada um.

Carlos Henrique C. Duarte — MG

## Linha ZX81

### INKEY\$ para várias teclas

Diversas vezes, num programa em BASIC, temos necessidade de interpretar mais de uma tecla, pressionada simultaneamente.

Para resolver tal problema, pode ser utilizada uma variável do sistema contida nos endereços 16421 e 16422, consequentemente, o valor das teclas pressionadas, será:

```
PEEK 16421+256=PEEK 16422
```

Basta guardar o resultado em uma variável e, a partir daí, fazer todo o resto. Veja um exemplo com este pequeno programa que interpreta apenas o pressionar simultâneo de 5 e 0 ou 8 e 0, muito utilizado em alguns jogos.

```
10 CLS
20 LET X=PEEK 16421+256=PEEK 16422
30 IF X=56807 THEN PRINT AT 10,16;"5
E 0"
40 IF X=62959 THEN PRINT AT 10,16;"8
E 0"
50 GOTO 10
```

Os valores das teclas pressionadas podem ser obtidos a partir da seguinte rotina:

```
9000 PRINT AT 21,0: PEEK 16421+256=PE
EK 16422
9010 GOTO 9000
```

## Linha TK 2000

### Chamadas em Assembler

Duas dicas para os novos usuários do TK 2000 que acham que call's no TK 2000 não podem ser usados com os do Apple. É um engano. Por exemplo, digite a seguinte rotina:

```
50000 DATA 160,1,162,0,138,24,23
3,1,208,252,141,48,192,232,224,2
55,208,242,236,200,237,96
50010 FOR A=768 TO 769: READ B: PO
KE A,B: NEXT
50020 FOR A=1 TO 255 STEP 3: CALL
7AH: NEXT
```

E veja que, tanto no Apple quanto no TK 2000, ela produz um interessante efeito sonoro. Portanto, POKES a partir do endereço 768 rodam perfeitamente no TK 2000.

Para simular o CALL -958 do Apple, basta substituí-lo por PRINT SPC (39): HTAB 1. Os CALL -936 do Integer BASIC de alguns Apple II, também rodam perfeitamente no TK 2000, e sem modificações.

Bruno Lopes Fernando Cabral — PB

Para estudar FORTH, é preciso conhecer, pelo menos um pouco, sobre programação estruturada. Nesta quarta lição, você vai aprender algumas estruturas de controle que FORTH oferece.

# Curso de FORTH (IV)

Antonio Costa

Quando FORTH foi inventado, um modismo denominado programação estruturada estava entusiasmando os cientistas de computação. Este modismo influenciou de tal forma as pessoas que conceberam FORTH que se tornou impossível estudar a linguagem sem saber o que é programação estruturada.

Existem trechos de programas que, dependendo da situação, podem ser ou não executados. Outros trechos devem ser executados repetidamente enquanto certas condições forem satisfeitas. Isso significa que toda linguagem precisa ter comandos para controlar a execução dos outros comandos. Tradicionalmente, tal controle era utilizado pelo IF...THEN GOTO. Foi então que o holandês *Dijkstra* escreveu um artigo denominado "GO TO Considerado Prejudicial". Neste artigo, ele propõe que se evite o uso de GO TO em certas circunstâncias. Para substituir o GO TO, foram propostas estruturas de controle fixas que no FORTH são representadas por IF ELSE THEN, BEGIN UNTIL e BEGIN WHILE REPEAT. Vamos dar uma olhada em cada uma delas.

## IF THEN ELSE

O funcionamento do IF pode ser descrito assim: uma comparação é realizada por palavras do tipo > (maior), < (menor) ou = (igual). Esta comparação é feita entre o segundo elemento da pilha e o topo, tendo como resultado os valores booleanos falso ou verdadeiro. Falso é representado por 0 e verdadeiro, por 1. Caso o resultado seja verdadeiro, todas as palavras entre IF e ELSE são executadas. Caso seja falso, as palavras entre ELSE e THEN são executadas.

Existe a possibilidade de IF ser usado sem ELSE. Quando isto ocorre, as palavras que estão entre o IF e o THEN devem ser executadas, se o resultado da comparação for verdadeiro.

A figura 1 resume o comportamento do IF. Vamos ver alguns exemplos que tornarão as coisas mais claras.

Digamos que eu queria definir uma palavra que imprima uma advertência, se o número que está no topo da pilha for maior que 100. Posso fazer isto com a definição abaixo:

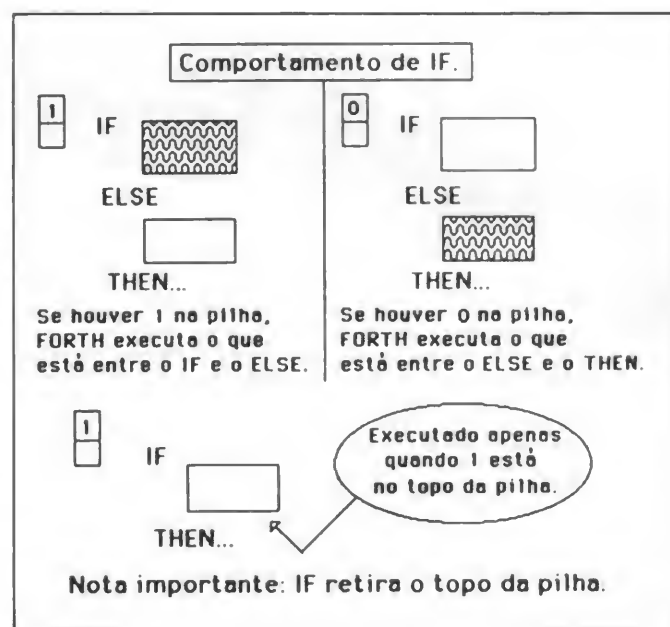


Figura 1

```
: ADV DUP 100 >
  IF ." NÚMERO MUITO GRANDE" CR
  THEN ;
```

O DUP serve para fazer uma cópia do número que está no topo da pilha, conforme mostrado no seguinte esquema:



A cópia é necessária porque as palavras comparadoras retiram da pilha os dois elementos que comparam e não queremos perder o número original. O ponto seguido de aspas imprime uma mensagem. No exemplo, esta mensagem é NÚMERO MUITO GRANDE. Observe que a mensagem termina com as

pas. Observe também que não existe espaço entre o ponto e a primeira aspa e que existe um espaço entre a primeira aspa e a mensagem. A palavra CR faz com que o computador mude de linha após a impressão da mensagem. Vejamos o funcionamento da palavra que acabamos de definir:

```
54 ADV OK
200 ADV
NÚMERO MUITO GRANDE OK
```

O número 54 não provocou nenhuma reação de ADV. O número 100 fez com que a mensagem fosse impressa.

Tentemos outro exemplo. Vamos definir uma palavra que substitui o topo da pilha por 1, se ele for maior que 0; por -1, se ele for negativo e por 0, se ele for nulo. Esta palavra é chamada de sinal pelos matemáticos.

```
: SINAL DUP 0 >
  IF DROP 1
  ELSE 0 =
    IF 0
      ELSE -1
    THEN THEN ;
```

Deixo para o leitor a decifração do funcionamento de SINAL.

É boa prática de programação colocar os componentes de uma estrutura de controle um embaixo do outro. Isto pode ser visto na definição de SINAL: os ELSEs, estão embaixo dos IFs, e os THENs, dos ELSEs.

## BEGIN UNTIL

O comportamento de BEGIN UNTIL pode ser resumido assim: entre o BEGIN e o UNTIL existem várias palavras que são executadas repetidamente até que o UNTIL encontre a indicação de verdadeiro na pilha. Este comportamento é esquematizado na figura 2. A indicação de verdadeiro, conforme já vimos, é o número 1.

Vamos usar BEGIN UNTIL para definir uma palavra muito útil. Trata-se da SN. Esta palavra lê o teclado até que uma pessoa tecla S (simbolizando sim) ou N (representando não). Caso a letra teclada seja S, 1 é deixado na pilha. Caso seja N, 0 é deixado na pilha. Veja o exemplo:

```
: SN BEGIN ." Tecla S(im) ou N(ão)" CR
  KEY DUP 78 =
  IF DROP 0 1
  ELSE 83 =
    IF 1 1
    ELSE 0
  THEN THEN
  UNTIL ;
```

Os números 78 e 83 são os códigos de S e de N. Se a letra teclada for N (código 73), será executado "IF DROP 0 1". O 1 servirá para indicar ao UNTIL que ele pode parar de pedir à pessoa que

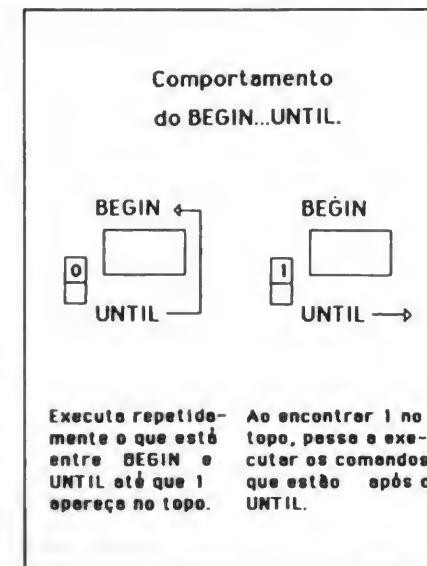


Figura 2

tecle S ou N. O 0 indica que N foi teclado. Se a tecla pressionada não for N, ela é comparada com S (código 83). Caso ela seja a S, FORTH executará "IF 1 1". O primeiro 1 serve para mostrar que S foi apertado; e o segundo, para fazer o UNTIL interromper as repetições. Se a pessoa não apertou nem S nem N, 0 é deixado na pilha e, ao ser encontrado por UNTIL, fará com que o programa peça mais uma vez que o usuário tecla S(im) ou N(ão). Nosso programinha é teimoso: enquanto alguém não aperta S ou N, ele não pára de insistir.

## BEGIN WHILE REPEAT

O BEGIN WHILE REPEAT não é muito diferente do BEGIN UNTIL. Entre o BEGIN e o REPEAT há várias palavras que devem ser executadas repetidamente enquanto o WHILE encontrar um sinal de verdadeiro deixado na pilha por alguma comparação. Se WHILE encontrar um sinal de falso, ele interromperá as repetições. O exemplo que vamos fornecer é uma palavra que pede repetidamente números e os soma ao topo da pilha.

```
: SOMATÓRIA 0
  BEGIN ." DE-ME UM NÚMERO" CR #IN
  DUP 0 >
  WHILE REPEAT DROP ;
```

Inicialmente, um 0 é colocado no topo da pilha. Números são pedidos, enquanto a pessoa que os está teclando não entrar com um que seja negativo. A palavra que lê inteiros do teclado e os coloca na pilha é # IN.

## CARACTERES GRÁFICOS

A maioria dos computadores possui certos códigos que, impressos com

**CENTRALDATA**  
Com e Representações Ltda

## SUPRIMENTO É COISA SÉRIA

- DISKETES: 5 1/4 e 8" e fitas magnéticas - marca DATALIFE VERBATIM
- ETIQUETAS PIMACO - PIMATAB
- FORMULÁRIOS CONTÍNUOS E PASTAS
- FITAS P/IMPRESSORAS EM GERAL
- ARQUIVOS PARA DISKETTES

## PRODUTOS COM GARANTIA E ENTREGA IMEDIATA

AV. PRES. VARGAS Nº 482 GR. 201/203  
Tel.: KS (021) 253-1120  
Telex: (021) 34318

EMBLER (I) — Artigo de Evandro a a linha Apple.

Série de autoria do professor

53 LIVROS

54 DICAS



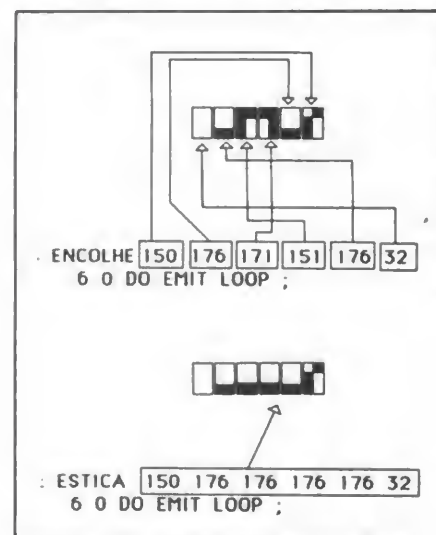


Figura 3

EMIT, produzem símbolos gráficos. É possível imprimir seqüências destes símbolos gráficos de modo a formar desenhos muito úteis na confecção de jogos. A figura 3 mostra uma das muitas maneiras de fazer isto. Nesta figura, há um esquema indicando como uma pequena minhoca egípcia pode ser desenhada tanto com o corpo encolhido

quanto com o corpo esticado. Os códigos gráficos usados para criar a minhoca são de máquinas compatíveis com o TRS-80. Caso sua máquina seja da linha Sinclair, consulte o manual para ver quais símbolos deve usar a fim de obter efeito semelhante.

Na figura 4 mostramos como movimentar a minhoca pela tela. Nesta listagem, a palavra PAUSA emite uma nota musical de certa duração e tem duas finalidades: produzir som e fazer FORTH perder tempo de modo que o movimento da minhoca não seja muito rápido. Observe que usamos a palavra NOTE, cuja definição foi dada na primeira lição. Reutilizar palavras definidas anteriormente é muito popular entre programadores FORTH, pois diminui em muito o tempo de desenvolvimento de aplicativos. Aconselho o leitor a manter uma biblioteca de palavras reutilizáveis.

Vamos agora discutir a definição de MINHOCA apresentada na figura 4. A palavra PAGE apaga a tela. A seqüência "15 EMIT" elimina o cursor de máquinas como o TRS-80 e o CP 500. Caso sua máquina não seja uma destas duas, não é preciso incluir "15 EMIT" na definição. A seqüência "6 I PTC ESTICA" coloca o cursor na coluna I da linha 6 e, em seguida, desenha a minhoca esticada.

```

: ENCOLHE 150 176 171 151 176 32
6 0 DO EMIT LOOP ;
: ESTICA 150 176 176 176 176 32
6 0 DO EMIT LOOP ;
: PAUSA 98 40 NOTE ;
: MINHOCA PAGE 15 EMIT
55 0 DO 6 I PTC ESTICA PAUSA
6 I PTC ENCOLHE PAUSA
LOOP ;

```

Figura 4

Como I está variando de 0 a 53, haverá a ilusão de movimento.

Gostaria de chamar a atenção do leitor para o fato de que o último caráter do corpo da minhoca é um espaço em branco (código 32, no TRS-80). Se este espaço em branco não for incluído, a minhoca deixará um rastro ao se movimentar.

Antonio Eduardo Costa Pereira é formado em Engenharia Eletrônica pela Escola Politécnica da USP. Fez mestrado em Ciência Especial no Instituto de Pesquisas Espaciais, em S. J. dos Campos, e doutorado em Engenharia Elétrica na Cornell University, de Nova Iorque.

## SERVIÇOS

## problema

Seu problema  
☐ **MS list**  
 Cr\$ 12.000  
 pag. Valor

## Projeto MICRO BUG

Sim, desejo receber

☐ a fita MICRO BUG, com cartão de referência, pela qual pagarei Cr\$ 40.000 + Cr\$ 7.800 pelas despesas do correio.

## Nºs atrasados

Sim, desejo receber

☐ os exemplares \_\_\_\_\_ de MICRO SISTEMAS (do 31 ao 39) pelos quais pagarei o preço unitário de Cr\$ 3.000.

n.º \_\_\_\_\_ no valor de Cr\$ \_\_\_\_\_ à ATI Editora.

Cargo \_\_\_\_\_ Empresa \_\_\_\_\_

Tel.: \_\_\_\_\_

Estado \_\_\_\_\_

ASSINATURA

00030

Rua Oliveira Dias, 153 - CEP 01433  
 São Paulo/SP - Tel.: (011) 853-7758

## SOFT SPEED

GARANTIA TOTAL • MANUAIS INCLUSOS

JOGOS:  
 ESCOLHA 3  
 E PAGUE 2 !!!  
 E AINDA TEM  
 BRINDE

## CP-400 JOGOS EM FITA

F101 - DECATHLON - O jogo mais famoso de todos  
 F102 - DOUBLE BECK - Preencha as imagens  
 F103 - DRACONIAN - Salve o astronauta no espaço  
 F104 - ICE HOCKEY - Hockey no gelo  
 F105 - MR. DIG - Fantástico e emocionante  
 F106 - INTERCEPT 4 - Destrua os inimigos galácticos  
 F107 - MISTAR - Jogo de aventura  
 F108 - SIMULADOR DE VOO - Excelente qualidade gráfica  
 F109 - POPEYE - Derrote o Brutus  
 F110 - GALAGON - Fantástico jogo  
 F111 - ZAXXON - O melhor jogo para micros  
 F112 - GERM - Defesa de dois germes  
 F113 - DEVIL ASSAULT - Emocionante!  
 F114 - AIR TRAFFIC CONTROL - Controle o aeroporto  
 F115 - CRASH - Defesa de dois insetos  
 F116 - SPACE WRECK - Defesa sua base espacial  
 F117 - SHAFT - Destrua os pontos e muito cuidado!  
 F118 - DESERT RIDER - Corrida de carros no deserto  
 F119 - CHOPPER STRIKE - Aventura com helicóptero  
 F120 - TIME PATROL - Salve os para-queixas

PREÇO DE CADA JOGO: Cr\$ 35.000

PARA CADA 2 JOGOS COMPRADOS, ESCOLHA UM TERCEIRO INTEIRAMENTE GRATIS. E AINDA GANHE MAIS UM JOGO: BLACKJACK.

## CP-400 APLICATIVOS E UTILITÁRIOS EM DISCO

D121 - VIPCALC - Planilha eletrônica 250.000  
 D122 - VIPWRITER - Processador de textos 250.000  
 D123 - VIPDATABASE - Banco de Dados 250.000  
 D124 - VIPTERM - Transforma o micro em terminal 250.000  
 COLEÇÃO VIP (4 PROGRAMAS ACIMA) 800.000  
 D125 - PRO COLOR FILE - Banco de dados 200.000  
 D126 - ELITE FILE - Banco de dados 200.000  
 D127 - ELITE CALC - Planilha eletrônica 200.000  
 COLEÇÃO ELITE (2 PROGRAMAS ACIMA) 300.000  
 D128 - DYNACALC - Planilha eletrônica 200.000  
 D129 - DYNAGRAPH - Utilitário para gráficos 200.000  
 COLEÇÃO DYNA (2 PROGRAMAS ACIMA) 300.000  
 D130 - TELE WRITER 64 - Processador de textos 400.000  
 D131 - GRAPHICON COMPLETO - Super utilitário gráfico 400.000  
 D132 - SPEED-CONTABILIDADE COMPLETA 400.000  
 D133 - SPEED-CAADASTRO DE CLIENTES 400.000  
 D134 - SPEED-FOLHA DE PAGAMENTO 400.000  
 D135 - SPEED-CONTROLE DE ESTOQUES 400.000  
 COLEÇÃO SPEED (4 PROGRAMAS ACIMA) 1.000.000  
 D136 - KIT EDITASM E MACROBLOC - Assemblagem 200.000  
 D137 - KIT SYNTHET 77 E MUSIC II - Soft/Música 200.000  
 D138 - PASCAL - Linguagem científica 300.000  
 D139 - FORTRAN - Linguagem científica 300.000  
 D140 - FORTH - Linguagem científica 300.000  
 COLEÇÃO DAS TRÊS LINGUAGENS ACIMA 700.000  
 D141 - PIRATECTOR - Proteja seu programa contra pirataria e coloque capas apresentação 400.000  
 D142 - CLONE - Copia discos protegidos 400.000  
 D143 - CLONE MASTER - O melhor copador discos 500.000  
 D144 - COLOR KIT - Copia disquetes e fitas de vídeo 300.000

D145 - TAPEUTIL - Copia disquetes e fitas de vídeo 300.000  
 D146 - DISK UTILITY - Copia, backup, desinstala, reinstala programas, lista arquivos, etc. 200.000  
 D147 - DISK DRIVER ANALYSER - Análise hardware 200.000  
 D148 - COLOR QUAYVER - Super sintetizador de som 300.000  
 D149 - TITULADOR PARA VT - Gerador caracteres 300.000

BRINDE: NA COMPRA DE QUALQUER PROGRAMA EM DISCO PARA CP400 GRATIS UM SUPER SINTETIZADOR DE VOZ

## CP400 UTILITÁRIOS EM FITA

F150 - SUPER SCREEN 51 - 24 Linhas texto/graf. 60.000

## JOGOS EM FITA PARA TX90X

F201 - XADREZ - Em sua melhor versão  
 F202 - REVERSI - Famoso jogo Othello  
 F203 - GAMÃO - Jogo de tabuleiro  
 F204 - ANT ATTACK - Destrua as formigas  
 F205 - CYBER RATS - Cuidado com os ratos!  
 F206 - HORACIO E AS ARANHAS - Um grande jogo!  
 F207 - 3D TANK - Batalha com tanques  
 F208 - SIMULADOR DE VOO - Impressionante!  
 F209 - COOKIE - Ajude o cozinheiro  
 F210 - HORACIO ESQUAVANDO - Aventura divertida  
 F211 - PYRAMARAMA - Tente vencer este desafio  
 F212 - ANDROID 2 - Sensacional!  
 F213 - PLANETOIDS - Excitante missão espacial  
 F214 - MISSILE - Cuidador missil fatal  
 F215 - PSST - Destrua os insetos  
 F216 - MARIO MINER - Conquiste a mina nas cavernas  
 F217 - HUNGRY HORACE - Cuidado com os quadros  
 F218 - PINBALL - Truque de fliper  
 F219 - SPACE RAIDERS - Defesa sua planície  
 F220 - ASTRO BLASTER - Cuidado com os invasores  
 F221 - JET SET WILLY - Tente fazer Willy correr  
 F222 - GHOST HUNT - Sensacional versão do Pacman  
 F223 - CHECKERED FLAG - Puxe um F1 de verdade  
 F224 - TENNIS - Excelente versão  
 F225 - ROBOTICS - Destrua os robôs invasores!  
 F226 - GALACTIC ABDUCTORS - Cuidado! perigo!  
 F227 - SINUCA - Tente matar todos os bolas  
 F228 - MINED OUT - Cuidado com as minas  
 F229 - MR. WIMPY - Pegue os ingredientes  
 F230 - GOLFE - Excelente e real jogo  
 F231 - CROSS - Tente atravessar a sua perigosa  
 F232 - JUMPING JACK - Cuidado com as fendas no chão  
 F233 - GROUND ATTACK - Penetre na caverna inimiga!  
 F234 - MONSTER MINE - Cuidado com os monstros da mina  
 F235 - SEIDAB ATTACK - Proteja sua cidade das aves  
 F236 - GALAXIANS - Destrua os invasores

PREÇO DE CADA JOGO: Cr\$ 35.000

PARA CADA 2 JOGOS COMPRADOS, ESCOLHA UM TERCEIRO INTEIRAMENTE GRATIS. E AINDA GANHE MAIS UM JOGO: SENHA.

## APPLE II - JOGOS ESPECIAIS EM DISCO - CADA Cr\$ 120.000

D301 - KARATECA - Lute na arte da verdade  
 D302 - CONAN - Aventura igual ao filme  
 D303 - GHOSTBUSTERS - Aventura igual ao filme  
 D304 - SUMMER GAMES - 8 Modalidades esportivas  
 D305 - CPI - Jogo pomográfico para adultos  
 D306 - SKYFOX - Puxe um caça de verdade

## APPLE II - JOGOS EM DISCO DA PROMOÇÃO

D307 - AZTEC - Destrua a pirâmide azteca  
 D308 - PICADILLY - Sensacional jogo  
 D309 - SHIFT SAM - Fantástico e emocionante jogo  
 D310 - DONKEY KONG - Mario contra o gorila  
 D311 - FREE FALL - Tente vencer este desafio  
 D312 - PIEMAN - Excelente e divertido  
 D313 - ORBITRON - Entre em órbita com seu micro  
 D314 - QIBERT - Tente se movimentar nos cubos  
 D315 - JUNGLE HUNT - Aventura na selva  
 D316 - CAVERNS OF CALISTO - Aventura  
 D317 - PITFALL II - Nas cavernas perdidas  
 D318 - TALIMAN - Labirinto tipo Pacman  
 D319 - XADREZ - Tradicional jogo  
 D320 - GAMÃO - Jogo milenar de tabuleiro  
 D321 - BUCK ROGERS - Puxe uma espaçonave  
 D322 - LOAD RUNNER - Pegue os tesouros

PREÇO DE CADA JOGO: Cr\$ 65.000

PARA CADA 2 JOGOS COMPRADOS, ESCOLHA UM TERCEIRO INTEIRAMENTE GRATIS. E AINDA GANHE MAIS UM JOGO: POKER

## APPLE II - UTILITÁRIOS E APLICATIVOS EM DISCO

D351 - DISK DRIVE DIAGNÓSTICO 200.000  
 D352 - SAY IT - Sintetizador de Voz 200.000  
 D353 - MUSICOMP - Compositor de Música 200.000  
 D354 - KOALA - Criador de Desenhos 200.000  
 D355 - PRINT SHOP - Gerador de Impressos 170.000  
 D356 - THE COMPLETE GRAPHICS SYSTEM 200.000  
 D357 - SCREEN WRITER - Processador de Textos 200.000  
 D358 - SPEED-MALA DIRETA 300.000  
 D359 - SPEED-CONTROLE DE ESTOQUES 300.000  
 D360 - SPEED-CONTAS A RECEBER 300.000  
 D361 - SPEED-FOLHA DE PAGAMENTO 300.000  
 D362 - LOCKSMITH 5.0 - Utilitário de Cópia 170.000  
 D363 - MAGIC WINDOW - Processador de Textos 300.000  
 D364 - VISCALC II - Planilha Eletrônica 170.000  
 D365 - VISFILE II - Banco de Dados 170.000  
 D366 - DBASE II - Completo 500.000  
 D367 - FRIDAY - Completo 450.000  
 D368 - WORDSTAR - Processador de Textos 400.000  
 D369 - LOTERIA ESPORTIVA 170.000  
 D370 - LOTO 170.000

BRINDE: GRATIS, NA COMPRA DE QUALQUER PROGRAMA EM DISCO UM CURSO PASSO-A-PASSO EM BASIC.

## COMO COMPRAR:

NÃO RECORTE SUA REVISTA. FAÇA O SEU PEDIDO POR CARTA, RELACIONANDO SEUS DADOS E OS CÓDIGOS DOS PROGRAMAS DESEJADOS. ANEXE À CARTA UM CHEQUE NOMINAL À ATI Editora Ltda., Av. Pres. Wilson, 165 grupo 1210 - Centro, CEP 20.030 - Rio de Janeiro - RJ. As despesas do correio já estão incluídas.

SOFTSPEED - CAIXA POSTAL 40088 - RIO-RJ

rm  
 EDITORA

NOVOS  
 LANÇAMENTOS



OS 40 MELHORES JOGOS PARA SEU MICRO  
 Uma sensacional coleção de jogos de inteligência e ação incluindo jogos em linguagem de máquina. Código 200.1. Preço: Cr\$ 39.990.



INTRODUÇÃO BÁSICA AOS MICROCOMPUTADORES E A INFORMÁTICA  
 de Marcelo Menezes e Renato Audi. Livro que aborda de forma simples e clara a evolução dos computadores, linguagem de programação, atividades profissionais, entre outros assuntos. Código 300.2. Preço: Cr\$ 39.990.

NAS  
 MELHORES  
 LIVRARIAS  
 DO  
 PAÍS

PEDIDOS PARA ATI Editora Ltda., Av. Pres. Wilson, 165 grupo 1210 CEP 20.030 - Rio de Janeiro - RJ, anexando cheque nominal.



A Microdigital  
lança no Brasil  
o micro pessoal  
de maior sucesso  
no mundo.

A partir de agora a história dos micros pessoais vai ser contada em duas partes: antes e depois do TK 90X.

O TK 90X é, simplesmente, o único micro pessoal lançado no Brasil que merece a classificação de "software machine": um caso raro de micro que pela sua facilidade de uso, grandes

recursos e preço acessível recebeu a atenção dos criadores de programas e periféricos em todo o mundo.

Para você ter uma idéia, existem mais de 2 mil programas, 70 livros, 30 periféricos e inúmeras...



revistas de usuários disponíveis para ele internacionalmente.

E aqui o TK 90X já sai com mais de 100 programas, enquanto outros estão em fase final de desenvolvimento para lhe dar mais opções para trabalhar, aprender ou se divertir que com qualquer outro micro.

O TK 90X tem duas versões de memória (de 16 ou 48 K), imagem de alta resolução gráfica com 8 cores, carregamento rápido de programas (controlável pelo próprio monitor), som pela TV, letras maiúsculas e minúsculas e ainda uma exclusividade: acentuação em português.

Faça o seu programa: peça já uma demonstração do novo TK 90X.

# MICRODIGITAL

# Chegou o micro cheio de programas.

